Using a multimedia semantic graph for web document visualization and summarization

Antonio M. Rinaldi[1,2] (ID) · Cristiano Russo[1]

Abstract

The synthesis process of document content and its visualization play a basic role in the context of knowledge representation and retrieval. Existing methods for tag-clouds generations are mostly based on text content of documents, others also consider statistical or semantic information to enrich the document summary, while precious information deriving from multimedia content is often neglected. In this paper we present a document summarization and visualization technique based on both statistical and semantic analysis of textual and visual contents. The result of our framework is a Visual Semantic Tag Cloud based on the highlighting of relevant terms in a document using some features (font size, color, etc.) showing the importance of a term compared to other ones. The semantic information is derived from a knowledge base where concepts are represented through several multimedia items. The Visual Semantic Tag Cloud can be used not only to synthesize a document but also to represent a set of documents grouped by categories using a topic detection technique based on textual and visual analysis of multimedia features. Our work aims at demonstrating that with the help of semantic analysis and the combination of textual and visual features it is possible to improve the user knowledge acquisition by means of a synthesized visualization. The whole strategy has been evaluated by means of a ground truth and compared with similar approaches. Experimental results show the effectiveness of our approach, which outperforms state-of-art algorithms in topic detection combining both visual and semantic information.

✉ Antonio M. Rinaldi
antoniomaria.rinaldi@unina.it

Cristiano Russo
cristiano.russo@unina.it

[1] Department of Electrical Engineering and Information Technologies, University of Naples Federico II, Napoli, Italy

[2] IKNOS-LAB Intelligent and Knowledge Systems (LUPT), Napoli, Italy

Springer

1 Introduction

The fast growth of user-centered information retrieval applications required new method-
ologies and techniques to assist users during their searching and browsing processes. In
this scenario, people use different tagging services to manage, organize and retrieve useful
information. A tagging system does not require much effort by the user and it is a good way
to find relevant information. Nowadays a user inserts tags into videos, images and other
resources with just a few words to easily retrieve and share them. User-centered approaches
need effective methodologies to implement efficient strategies for the cooperation and anal-
ysis of data and applications, supporting the use of formal knowledge representations such
as ontologies and semantics [62]. In this context, new techniques have been proposed in
recent years to help users in the representation of synthesized information. One of these
approaches is based on the creation of tag clouds. They are visual representations of a set
of terms in which the text properties (e.g. size, color and weight) are used to indicate the
relevant properties between words and documents. They can be organized according to dif-
ferent characteristics: (i) a tag for the frequency of each object; (ii) a global tag cloud where
the frequencies are aggregated over all users and objects; (iii) a cloud containing categories
where the size shows the number of subcategories.

Tag clouds derive from the collaborative tagging paradigm [26, 45] used in many social
networks, forums and communities where users play a central role [49]. In these systems,
users annotate the contents with free keywords (tags) without having to use predefined
ones. This classification system is called *Folksonomy* [69], which appears to be in contrast
with other types of classifications (e.g. ontologies and thesaurus). This kind of visualization
involves less physical and cognitive workload than thinking about a search tag that defines
the thematic field to be explored. For example, after an initial tag is set, users can browse
using related tags. Social networks have greatly increased the popularity of this type of visu-
alization, but various problems arose, related to their basic structure and to the completely
free and out of control choice for tags. As a matter of fact, it is difficult to get a consistent
view of the tags used in the whole system, since it is common to observe terminological and
conceptual heterogeneities between users. In addition, the folksonomies systems user inter-
face does not support users searching for appropriate tags and creating efficient queries for
discovering interesting content. As discussed in [29] and [4] if visible tags are selected only
according to their frequency there could be problems of high semantic density because few
topics will tend to dominate the entire visualization and less important terms will vanish.

Tag clouds can be used to synthesize document collections by providing a summary of
their information content and this structure can be a useful tool in the browsing process
[60]. Generally speaking, a text document contains only text, while multimedia documents
contains both text and other information, such as images, audio, video, etc. The purpose of
summarization is to reduce the length of a document by creating a summary that preserves
the most important contents of the original document. Generally speaking, summarization
methods extract the most important information content, i.e. most representative sentences
or keywords in the documents, and present it to the user in a small space. Existing meth-
ods use the information contained in the analyzed document based on natural language
processing (NLP) or statistical techniques. A more detailed survey of techniques used for
text documents summarization is provided in [42]. In general, there are two approaches: (1)
Extraction - it recognizes a subset of terms or phrases in the original document to create a
summary; (2) *Abstraction* - it constructs an internal semantic representation using natural
language generation techniques to create a synthesis closer to what a human could generate.

Statistical information, like term frequency and word histograms are used to extract the terms to be added in the summary. Instead, in the processes of natural languages (NLP) [74], relevant information is related to a specific knowledge base using different techniques (ontologies, linguistics, semantic relatedness metrics, rule-based approaches, etc.). Existing document summarization techniques are based on text content of document collections, hence the resulting summary for documents is often misleading, or incomplete. Besides, methods relying on the mere text analysis are negatively affected by the issue called *polysemy*, that is the property of terms to have multiple meanings. In our vision, the semantic enrichment of most important document terms, together with the exploitation of multimedia representations provided with documents, are basic "tools" needed to solve or at least mitigate these problems. We define as *generic summary* a general summary of a document without further suggestions. This means that the summary is generated from the original document considering it as a whole unit. A *query relevant summary* instead, is the summary that is related to a given query or topic, that is the summary generated giving more importance to the informative content specified in the user query.

In addition to summarization, in this work we are also interested into the automatic use of *topic detection* and *categorization* tasks to allow a quick access to document collections. One disadvantage of many classification methods is that they use a flat categorization structure, regardless of the relationships between categories. On the other hand, hierarchies or taxonomic structures are the preferred ways in which concepts, subjects, or categories are arranged in practice [54]. They offer clear advantages in supporting activities such as information browsing, searching or visualization [58].

In this paper we present a novel methodology based on the combination of statistical information and NLP techniques, focusing on the generation of a *generic summary* based on the extraction of keywords. Our approach presents several innovations related to the multimedia document analysis research field. We propose a new framework that adds semantically related terms to the keywords extracted from the analyzed documents using semantic analysis and metrics based on visual features. The idea of exploiting visual information extracted from digital images for the task at hand comes from the proven effectiveness of such kind of information for improving performances in several application domains (e.g., images denoising [35, 50] and dehazing [40], early diagnosis of leukemia [52] and detection of cerebral aneurysm [39], face recognition [27], perception in autonomous-driving systems [38, 44], augmented reality in sports [43]).

The additional information is extracted from a general multimedia knowledge base. The concepts in this knowledge base are represented according to various multimedia forms, like text, images, etc.. A semantic topic detection strategy based on the combination of a textual and visual analysis is used to categorize documents and to help users during the browsing processes. The result of our approach is a new tool for document visualization that allows a better understanding of information contents and a quick access to documents in a collection. Our framework has been extensively tested and evaluated comparing its performances with some state-of-the-art algorithms commonly used in document summarization and topic detection. We also show the advantage of using a combination of semantic and multimedia information. Moreover, we highlight the use of a formal multimedia semantic knowledge base implemented in a NoSQL DB for several tasks of our framework.

This work pursues the following objectives: it focuses on novel techniques for the generation of a generic summary of a document; it exploits a semantic multimedia knowledge base for improving the task of topic detection by combining both semantic and visual features.

The article is structured as follows: Section 2 is devoted to analyze the literature about existing methods and technologies for document visualization and summarization; Section 3 describes the proposed architecture, showing each individual component and explaining the used techniques and metrics; in Section 4 we discuss the strategy used to evaluate our framework and the obtained experimental results; Eventually, conclusions and new directions of our research are presented in Section 5.

2 Related works

In our approach we use a *keywords extraction technique* to build a summary through tag clouds. The techniques for keywords extraction can be divided into 4 categories:

– Statistical approach: the relevance of a term is identified through statistical properties (for example TF-IDF, N-Grams, etc.);
– Linguistic approach: syntactic, semantic and lexical properties together with similarity functions are used to identify relevance keywords;
– Machine learning approach: the keywords extraction is based on a trained model used to predict the most relevant keywords (Naive Bayes, Support Vector Machine, etc.);
– Other approaches: different combinations of other approaches together with additional information such as word position, feature layout, HTML tags, etc..

The approaches based on machine learning in addition to semantic information [34] and co-occurrence [46] were extensively used for extracting keywords from a single document. In [67] the authors make use of feature selection and classification for a semantic information retrieval based on ontologies. The CSO classifier [65] is focused on topic detection in scholarly articles using an ontology-driven approach. Instead, in [2] different information is used based on the vector space model and genetic algorithms to measure the score between the sentences and the weights associated to the features. A very similar approach in [3] is based on an unsupervised technique called MCMR (Maximum coverage and minimum redundant) which allows to extract the most relevant and non-redundant sentences from the original document. In [33] the authors use linguistic features to represent the importance of a term, also based on the position in the document. Zhang [75] proposes an improvement of the TF-IDF algorithm for text categorization based on class discrimination strength. In [53] term frequency, proportional document frequency and semantic information are used for topic detection and technology trend tracking for patents. A tag-oriented approach is discussed in [76] where the authors present a new algorithm that uses a linear transformation to estimate the importance of tags. The latter are extended by adding related words through data mining techniques. In [72] new features are proposed for the extraction of keywords and the generation of a title based on Wikipedia as a knowledge base. An approach to synthesize selected text in a document is presented in [7]. The authors proposed a solution based on an algorithm for extracting keywords belonging to a context or language. An iterative approach to extracting keywords based on relationships between different granularities (words, sentences or topics) is presented in [70]. This method builds a graph that represents the relationships between different nodes and then through an iterative algorithm calculates the score of keywords. In [36] the author describes an application in which word clouds are used to browse and synthesize Twitter results with the aim of showing a first opinion of the contents of the tweets. In [31] the authors propose a new multi-document summarization system called FoDoSu (Folksonomy-based Multi-Document Summarization) based on

social folksonomies. The word frequency table is created for semantic analysis and the relevance of words is measured using HITS algorithms. Afterwards, using the clusters tags, the system analyzes the semantic relations between words and the word frequency table. Moreover, several studies have presented proposals for adding information to folksonomies based on different techniques [4]. In [24] a metric to measure tags similarities using statistics is proposed. Cluster-based algorithms have also been proposed, for example in [29] where the k-means was used to group semantically similar tags. Several approaches tackle the topic detection issue from a semantic point of view. Chen [20] uses Markov decision processes for a semantic-based topic detection. Hernandez [30] employs semantic techniques for semantic topic detection focusing on the Spanish language. Other examples of semantic topic detection and information retrieval are proposed in [16, 28, 64]. Another method used for the construction of a semantic network based on the co-occurrences of tags with the aim of comparing the structure of the folksonomies network is proposed by [15]. The authors have analyzed the similarities between tags and documents to enrich semantic and hierarchical aspects. In [47] the authors present a model for the combination of ontologies and social networks with the use of tag based techniques. Gemmell [25] proposed an algorithm for the construction of hierarchical structures starting from folksonomies designed to provide customized navigation processes. In [68] a tag clouds based interface was created to support user searching process. It has been observed that the scanning process and the click on tags is easier than the formulation of a query. In [21] the authors present a methodology for collaborative semantic understanding of generated tags. They proposed a visualization called TagClusters in which the tags are grouped into different semantic groups and the visual distance represents the semantic similarity between tags.

The main differences between the proposed method and the other ones presented in literature are in the use of a general knowledge base to represent the domain of interest of the users and in a completely automated process without a learning step. Moreover, we propose a different strategy based on the dynamic extraction of semantic networks from the knowledge base and we use a metric for the generation of tags based on a combination of statistical and semantic properties. Current state-of-art approach are mostly based on textual information retrieval and topic detection, but they often neglect visual information present in multimedia rich documents which could be extremely helpful in document summarization and categorization task. One of the goal of our work is to effectively exploit such visual information. In fact, other information obtained from the semantic network analysis is added to the classic tag clouds and different (multimedia) representations of the same concept are shown in addition to lexicon-semantic information. The design and implementation of a semantic multimedia knowledge base is a basic steps in several tasks of our framework related to knowledge representations and analysis giving us a formal interpretation of document topic. By means of the proposed knowledge base, it is able to classify multimedia documents (containing text and images) and group them into semantic categories, through a textual and visual analysis.

3 The implemented system

In this section we present the system architecture describing in detail all modules and their implementation. The proposed system follows a standard architecture usually designed for classical information retrieval systems, properly reorganized and customized in order to meet our goals. A specific emphasis is placed on the core of our framework, that is the process of building semantic tag clouds, describing in details techniques and technologies used

in the macro-block *Semantic Tag Cloud (STC) Builder*. The whole architecture is shown in Fig. 1 and described below in detail.

From a general point of view, a user can perform his search through a graphical interface created by the *Query Interface* module. The *Fetcher* aims to find documents contained in the *Document Collection* that are relevant for the submitted query. This collection is built from a repository (i.e. *Document Repository*) which, after a pre-processing phase during which the original documents are cleaned by the *Document Pre-Processor*, they are sent to the *Indexer*. It is in charge of storing and indexing the contents in order to optimize the search process. Cleaning operations are: (i) HTML tags removal; (ii) stopwords elimination; (iii) special characters cleaning, (iv) term stemming. The cleaned documents are sent to the macro module *Semantic Tag Cloud Builder* whose task is to provide a textual and visual representation of the main topics of the retrieved documents by means of the Semantic Tag Cloud Category (*STCC*) and the *Topic Detection (TD)* modules. The topic detection task is performed using textual (*TD Textual*) and visual (*TD Visual*) features with an ad-hoc combining function (*Combiner*). The use of semantic information is a basic step to improve the quality in recognizing main terms in the document and related concepts [55].

The text topic detection process is based on two modules, the *Word Sense Disambiguation (WSD)* which associates the right sense (meaning) with polysemic terms based on a semantic analysis using the *General Knowledge base*, and the *Semantic Network Extractor* able to dynamically extract a semantic subnet from the knowledge base. When the user selects a category from the *Semantic Tag Cloud Category (STCC)* it will correspond to a list of documents that will be shown with the *Semantic Tag Cloud Summaries (STCS)*.

The architectural components are further explained in the following paragraphs, providing for each of them more details about their structure and properties.
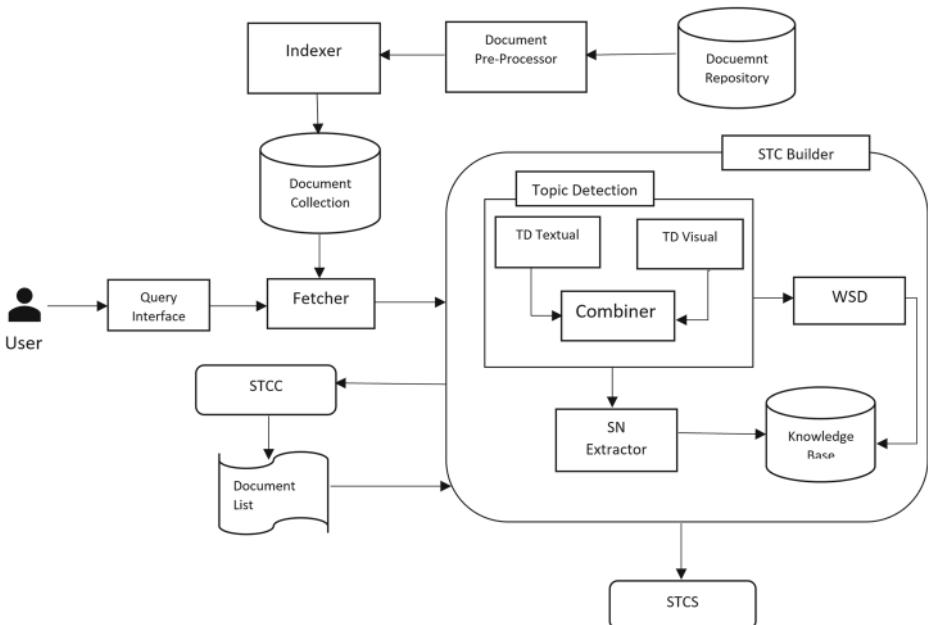

Fig. 1 System architecture

3.1 User query interface

It is the component that enables the interaction and communication of the system with the user. This component is devoted to user assistance through search and navigation tasks. The proposed interface allows the formulation of the query and the communication with the system back-end. The results visualization is a fundamental point of our approach and for this reason our interface shows the proposed Visual Semantic tag Cloud used to synthesize the retrieved documents that can also satisfy the user query [56].

3.2 Document pre-processing

This component consists in a set of tasks that are used to clean the original documents. The modules of this component are shown in Fig. 2.

Documents stored in the *Document repository* pass through several modules which are in charge of removing unnecessary parts from documents before indexing them into the *Document collection*. The first task is performed by the *Remove Tags* module. It consists in the removal of all tags and structures with respect to the document format (HTML, JSON, XML, etc.). In this process only the content (text and related images) is retrieved. The second module is named *Delete Stop-Words*. During this step the textual part of the content is processed for the deletion of stopwords - common words of a text that do not have a particular meaning in relation to the specific topic of the document (articles, prepositions, negatives, etc.); after this step, the module *Delete Special Chars* is in charge of deleting special chars from documents - characters or words that are not a part of the vocabulary (carry, tab, semicolon, comma, etc.). The last step of document pre-processing performs classical operations on text documents, i.e. *Stemming* and *Lemmatization*, to further reduce words diversity and to reduce the index size. Moreover, the text is converted in lowercase to avoid duplication of keywords in the dictionary generated by the indexer.

3.3 Fetcher

This component is in charge of collecting documents that satisfy the user information needs. Given a user query, the functionalities provided by the document pre-processing are used: (i) elimination of the stopwords, (ii) elimination of the special characters (iii) stemming with the lemmas. At the end of these steps the *tokenization* task is performed to extract the *keywords* from the query.
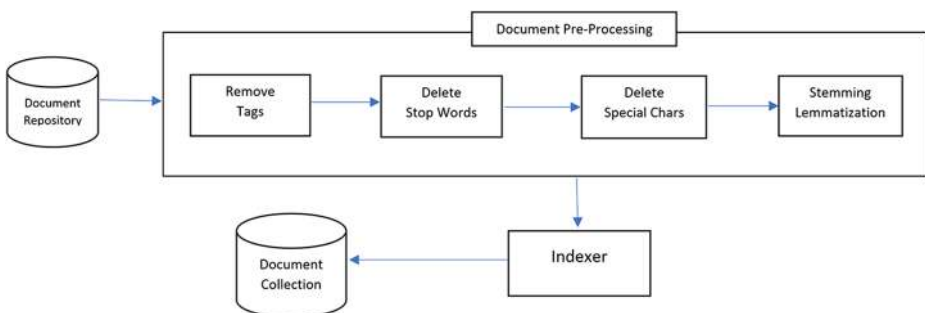

Fig. 2 Modular composition of Document Pre-Processing
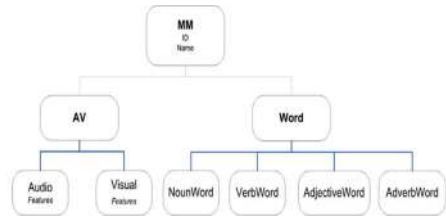
3.4 Knowledge base

The general knowledge base developed in this paper is based on a multimedia model pro-
posed in [57]. This formal representation uses *signs* which represent concepts, defined in
[22] as *"something that stands for something, to someone in some capacity"*. Generally
speaking, a concept can be represented in various multimedia forms such as text, images,
gestures, sounds or any other form through which information can be communicated as a
message. Each type of representation has some properties that distinguish them. The model
structure consists of a triple $< S, P, C >$ defined as:

- *S*: the set of signs;
- *P*: the set of properties used to relate signs with concepts;
- *C*: the set of constraints on the *P* collection.

In this work there are two types of representations, the *word* (i.e. text) and the *visual*
(i.e. images). The properties between representation and concepts are linguistic relations
and the constraints contain validity rules applied to properties with respect to the considered
multimedia representation. Knowledge is represented by an ontology implemented using
a semantic network (*SN*). It can be seen as a graph where the nodes are concepts and the
arcs are relationships between them. A concept is a set of multimedia data representing
an abstract idea. The language chosen to describe this model is the *description logic* (DL)
version of the *Web Ontology Language* (*OWL*), a markup language that offers maximum
expressiveness while retaining computational completeness and decidability. It allows the
declaration of disjointed classes as the property of a word to belong to a syntactic category.
It is also possible to declare the union classes used to specify domains, ranges and properties
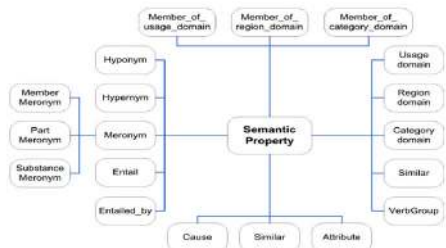to relate concepts and multimedia nodes.
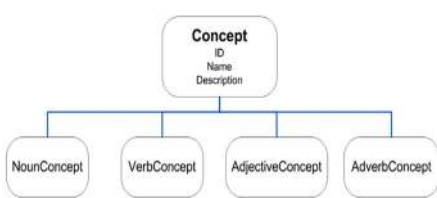
(a) Ontology model hierarchy for class Con- (b) Ontology model hierarchy for class MM.
cept.
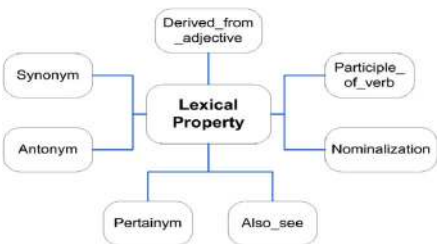
(c) View of lexical Properties   (d) View of semantic properties

Fig. 3 Concept, Multimedia and linguistic properties

Figure 3 shows the class hierarchies related to concepts, multimedia, lexical and semantic properties. The *MM* class with the respective sub-classes represents all the possible signs of our ontology. The classes have no elements in common and therefore are defined as disjointed. Table 1 shows some of the considered properties.

The attributes of Concept and Multimedia classes are also described. *Concept* has the following attributes: *Name* represents the name of the concept and the field *Glossary* contains a short description of it. The attributes of the *MM* subclass are: *Name* and *ID*. Each sub-class has its own set of features depending on the nature of the media. In this paper we use textual and visual representations and, in the *Visual* case, we consider the following global features:

- *Pyramid Histogram of Oriented Gradients* (*PHOG*) [8]: it counts the occurrences of the orientation of the gradient in localized portions of an image. This method is similar to the *Edge orientation Histogram* and to the *Scale-Invariant Feature Transform* but differs from the fact that it is calculated on a dense grid of evenly spaced cells and uses the overlapping local contrast normalization to improve accuracy;
- *Fuzzy Color and Texture Histogram* (*FCTH*) [19]: it is a low-level feature that combines color and texture in a single histogram. The histogram is generated by three fuzzy units: (i) the first unit is used to create a histogram with 10-bins where each bin (container) represents a predefined color, (ii) the second unit expands the histogram from 10-bins to 24-bins adding information from the hue of each color (hue), (iii) the last unit expands 24-bins into 192-bins. The size of this technique is limited to 72 bytes, appropriate for large databases;
- *Color and Edge Directivity Descriptor* (*CEDD*) [18]: it uses the same approach as the FCTH but its size is limited to 54 bytes compared to 72 of the FCTH. The difference lies in the third unit in which the 24-bins histogram is expanded into 144-bins using the 5 digital filters of the *Edge orientation Histogram* feature of the *MPEG-7* standard;
- *Joint Composite Descriptor* (*JCD*) [17]: this technique combines two CCDs descriptors, the CEDD and the FCTH, global descriptors that capture more than one feature at the same time with a very compact representation. It embeds color and texture information in a small histogram;
- *Color Layout* (*CL*) [37]: it represents the color distribution of visual signals in a compact form. This form makes the matching process efficient with low computational costs;
- *Edge Histogram* (*EH* [71]): it represents the distribution of 5 types of arcs, 4 directional and 1 non directional. This descriptor is *scale-invariant* and supports matching *rotation-invariant* and *rotation-sensitive* operations.

Table 1 Properties

Property	Domain	Range
hasMM	Concept	MM
hasConcept	MM	Concept
hypernym	Nouns and Verbs	Nouns and Verbs
holonym	Noun	Noun
Entailment	Verb	Verb
Similar	Adjective	Adjective

The linguistic properties are defined between terms (i.e. lexical properties) or concepts (i.e. semantic properties).

The considered lexical properties are:

– *Synonym*: it represents a relation between terms with the same meaning;
– *Antonym*: this property relate terms with opposite meaning;
– *Derivationally related forms*: it specifies that there exists a symmetric lexical morphosemantic relation between the first and second terms representing derivational morphology;
– *Nominalization*: it refers to the use of a word which is not a noun (e.g., a verb, an adjective or an adverb) as a noun, or as the head of a noun phrase, with or without morphological transformation;
– *Also see*: it specifies that additional information about the first term can be obtained by seeing the second term;
– *Pertainym*: it specifies an adjective word sense that pertains to either the noun or adjective second word sense or it specifies an adverb word sense that pertains to the adjective second word sense (Table 2).

On the other hand, the semantic properties defined in our model are:

– *Hypernym*: the generic term used to designate a whole class of specific instances;
– *Hyponym*: the specific term used to designate a member of a class;
– *Meronym*: the name of a constituent part of, the substance of, or a member of something;
– *Holonym*: it specifies that the second concept is a meronym of the first concept. This relation only holds for nouns;
– *Entailment*: it specifies that the second verb is an entailment of first verb. This relation only holds for verbs;
– *Cause*: it specifies that the second verb is a cause of the first verb. This relation only holds for verbs;
– *Domain*: it is a semantic organization of concepts with respect to specific properties (i.e. usage/region/category);
– *Similar*: a relation between adjectives with a similar meaning;
– *Member of*: it represents the relations of membership by means of specific properties (i.e. usage/region/category).

A systematic study on linguistics and the definitions of lexical and semantic properties are in [1].

The relations in the semantic network are represented as *ObjectProperties*. They depend on the syntactic category of the considered concept. Some of these properties (semantic or lexical) are shown in the Table 1 below:

Table 2 Properties constraints

Constraint	Class	Property	Constraint range
AllValuesFrom	Noun	Hypernym	Noun
AllValuesFrom	Adjective	attribute	Noun
AllValuesFrom	Verb	also see	Verb
AllValuesFrom	Noun	Hyponym	Noun

For example, the hypernymy property can only be used between nouns and nouns or between verbs and verbs. Each multimedia is linked to the represented concept by the ObjectPropety *hasConcept*, vice versa with *hasMM*. They are the only properties that can be used to link concepts with multimedia, the remaining properties are used to link multimedia to multimedia or concept to concepts.

The use of union classes give a precise meaning of properties range of application by means of domain and co-domain but at the same time do not exhibit perfect behavior in some cases. For example, the model does not know that a hyponymy property defined on sets of nouns and verbs would have a range of nouns when applied to a set of nouns and a range of verbs when applied to a set of verbs. To solve this problem we added some constraints and some of them are shown in the following table:

In some cases, the existence of a property between two or more individuals entails the existence of other properties. For example, if a concept A is a hyponym of a concept B, the concept B is hypernym of A. These characteristics are represented in OWL by means of property features. Some examples are provided in the following Table 3.

The ontological model has been translated into a Neo4J Graph database instance. The choice of this kind of NoSQL database is justified by its property to handle graphs and ontologies in a native way [10].

In fact, a knowledge conceptualization by ontologies has a logical representation using semantic networks and a physical implementation by means of a graph-oriented NoSQL DBMS.

3.5 Knowledge base implementation

The knowledge base used in our work has been implemented through the ontology model explained in the previous Section 3.4. It is a property-based graph model and it is populated with data from WordNet [48]. It defines and describes the concepts expressed by words and makes use of groupings of terms with a similar meaning, called *synset* (synonym set). Data information in WordNet is highly connected through various types of semantic and lexical relationships. In addition, we use ImageNet [23] for the visual representations. It is a database of images organized according to the hierarchy of WordNet (currently only for nouns) in which each node is represented by thousands of images with an average of about 500 images per node. From a volume dimension, the large size of WordNet and ImageNet in terms of number of nodes and relationships, the most obvious software solution is in the use of a NoSQL databases. Therefore, the considered ontology can be seen as a graph in which entities (concepts and multimedia) equal to nodes and ObjectProperties equal to relations between them. In our system we use Neo4J as database system to implement our graph. We will put in evidence the complexity of our knowledge base

Table 3 Properties features

Property	Features
hasMM	Inverse of hasConcept
hasConcept	Inverse of hasMM
hyponym	Inverse of hypernym: transitivity
hypernym	Inverse of hyponym: transitivity
verbGroup	Symmetry and transitivity

from both dimensional and heterogeneity points of view displaying its network implementation. In general we speak of macro visualization when we want to graphically represent the total or partial network, showing all nodes and interconnections between them, highlighting generic aspects regarding topology and complexity. On the other hand, when using the term micro-visualization we refer to a small graph with a limited number of nodes for recognizing specific or local aspects of a particular topological area. There are several tools for displaying graphs in Neo4j and some of most used are Cytoscape and Gephi [11].

For the macro view we decided to use Gephi, because it is a very flexible software for the customization of the retrieved data. In addition, it can be directly connected to Neo4j by using *APOC* procedures. Once the connection between the graph database and Gephi is created, the data is passed through Cypher queries [9].

In Fig. 4 a part of the implemented network is drawn, limited to 24271 nodes and 40000 relationships generated by a Cypher query that returns random paths. We have used *Force Atlas 2* as layout for this figure. In the following figure the resulting graph is shown together with the statistics offered by the tool.

The Figure is useful to highlight the complexity of the system, i.e. there is a high number of nodes and interconnections(arcs). Inspecting the statistics provided by the tool and the different colors used for different items, it is possible to detect how the hypernymy relationship outclass the others. Moreover, by setting a node size according to the incoming and outgoing arcs, it is possible to recognize nodes which are larger than others Such nodes represent more generic concepts and therefore have more lexicon-semantic links.

In Fig. 5 the number of nodes has been limited in order to display only the concepts direct connected by a single type of relationship. In this case we use the standard graphic Neo4j interface.

From this graph we can highlight, for example, how the ontology rules and constraints defined in the Section 3.4 are respected (inverse relation to another or constraints of type
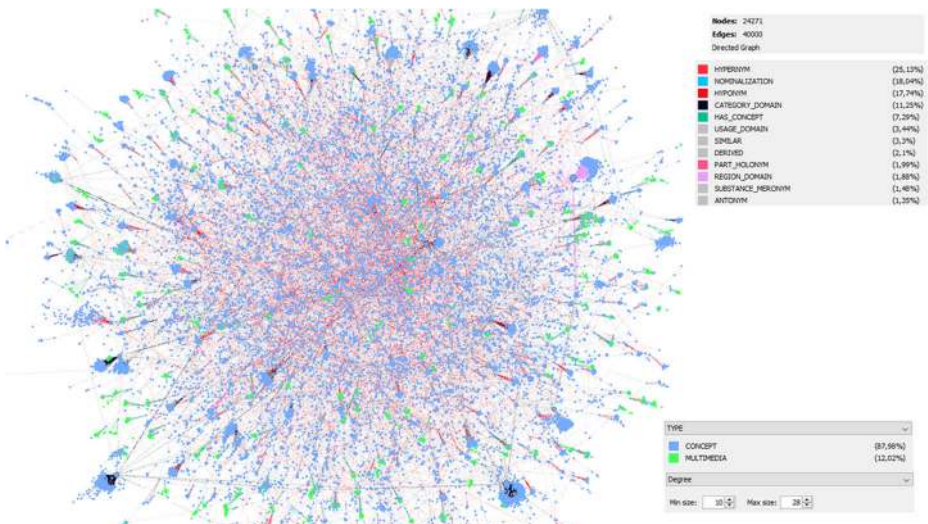

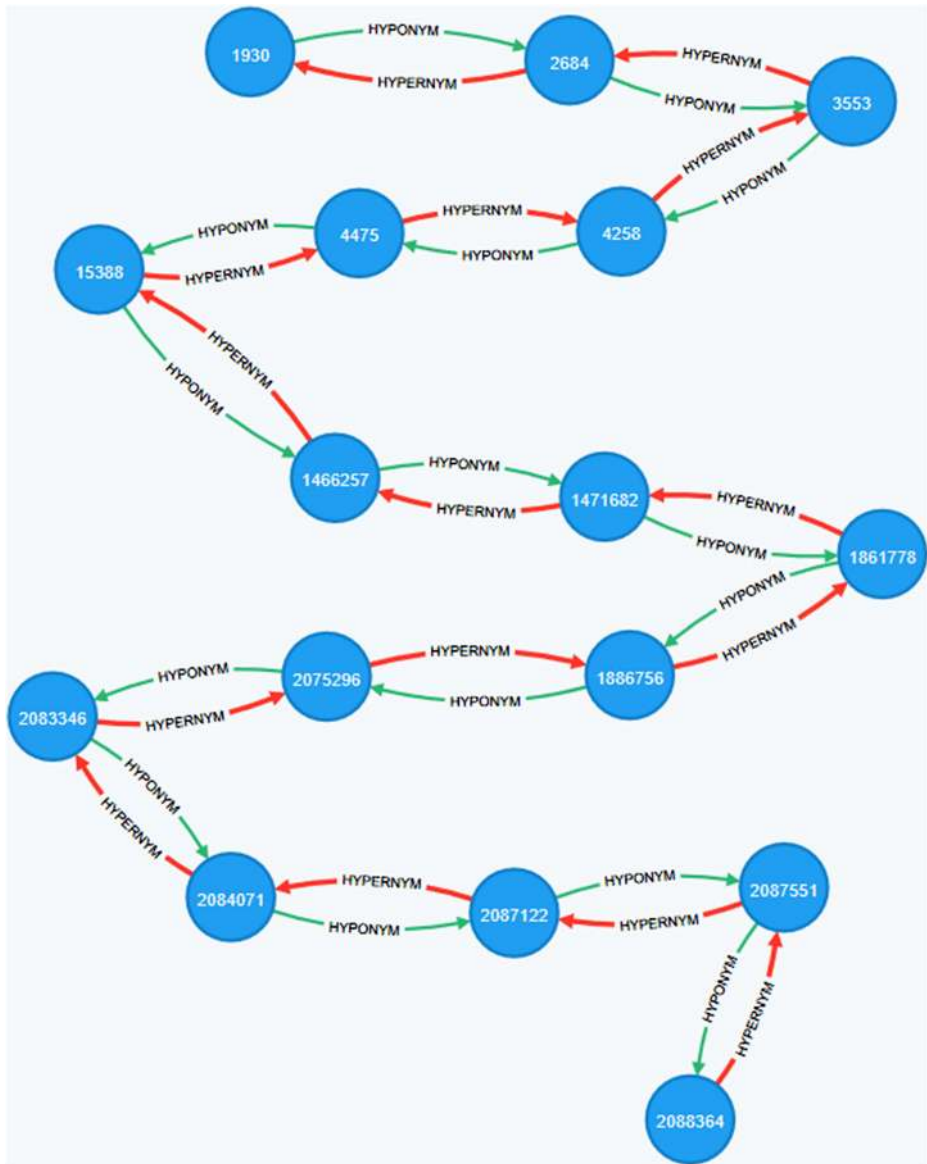**Fig. 4** Macro visualization of Neo4j db with Gephi

Fig. 5 Micro visualization of the semantic network

allValuesFrom). We can observe from the network topology that starting from any concept it is possible to easily retrieve not only the concepts directly connected to it but also other semantic link such as the hierarchical path composed of hypernymy relationships and hyponymy to the root.

In Fig. 6 a JavaScript extension of the web application provided by Neo4j is used to represent the visual nodes with the respective original images. This figure shows the concept *"spitz"*, a particular breed of dog, with its associated visual representations:
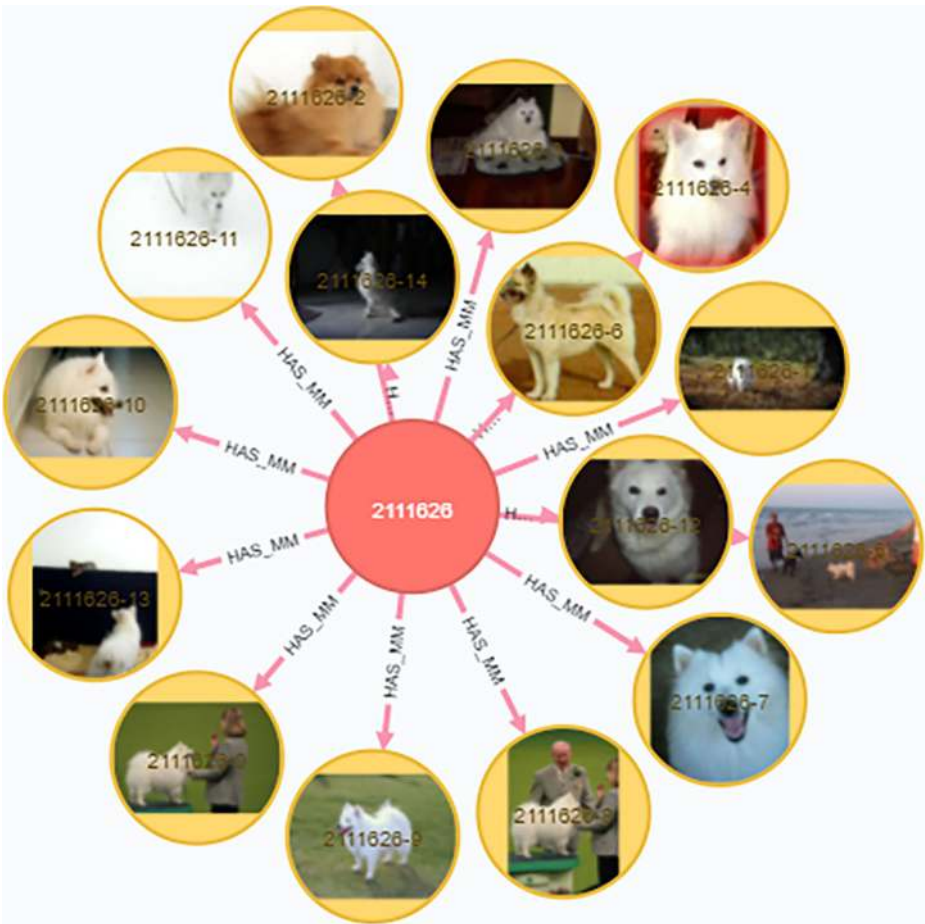
Fig. 6 Visualization of the concept and related multimedia

3.6 SN extractor

The aim of this module is to extract a subnet from the semantic network (i.e. our knowledge base), called *Dynamic Semantic Network* (*DSN*). The generated DSN is used by the *topic detection* module for the evaluation of the *sense coverage* that will be explained in the following subsection. The subnet is built starting from a concept recognized in the WSD step named *seed concept*. Moreover, all the concepts having a path consisting only of hyponym relations are added to the networks. In other words, the sub-graph generated is a hierarchy of concepts where more specific concepts are added as children of the root concepts in lower levels of the subnet. Figure 7 shows a subnet extracted from the knowledge base that has as its root the concept *working dog*, a hyponym of the *dog* concept. While Fig. 6 aimed to help the user in visualizing the multimedia items contained in the knowledge base, Fig. 7 is arranged as a dendrogram, that is a chart depicted as a tree. It shows how sub-concepts in the hierarchy are linked to the root concept as provided by the ontological model. On the left, we have the most generic concept, or root-concept (*working dog*). Moving from left to
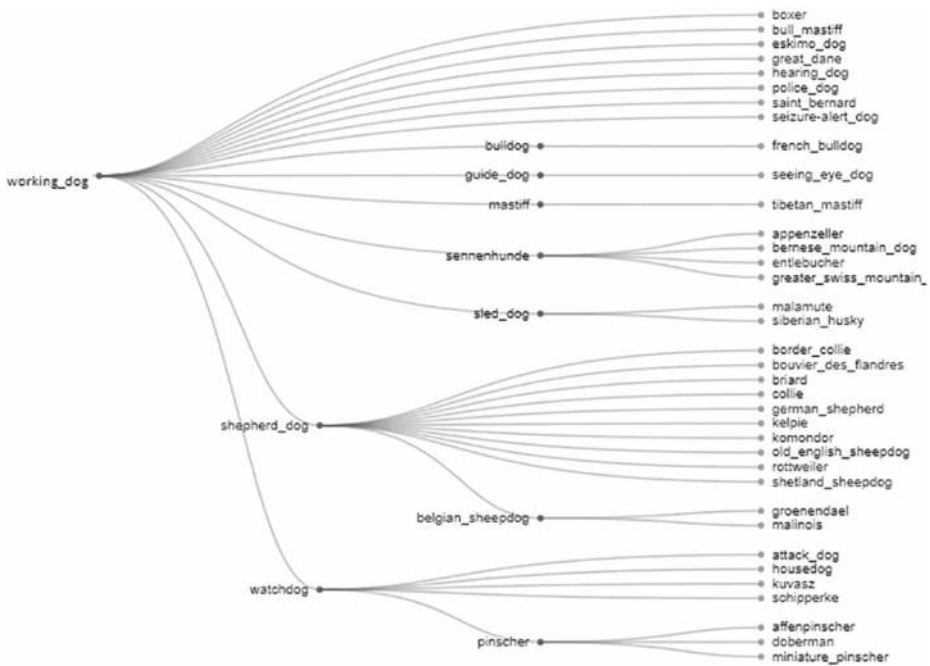
Fig. 7 SN Dendrogram of the concept *working dog*

right we have more specific concepts, where the leaves of the tree represent several breeds of dogs in this particular case.

3.7 Word sense disambiguation

A term can have different meanings due to a linguistic property called *polysemy*. The Word Sense Disambiguation (WSD) process assigns the right meaning to the analyzed terms based on the context in which they are found. The proposed technique consists in comparing each *sense* of a term representing one of the possible meanings with all the senses of the other ones. The similarity between terms is calculated using a linguistic-based approach and a new metric based on the measure of semantic relatedness. In our metric we assign a weight σ_i to the linguistic properties defined by the ontological model and implemented in our knowledge base, to express the strength of the considered relationship. The weights are shown in Table 4:

The weights were assigned to the relations considering the values proposed in [14, 54]. Weights are real numbers in the range [0,1] and these values have been assigned by experiments and validated by experts. The proposed metric is based on a combination of the path length between terms pairs, *best path*, and the *depth* of their nearest common subsumer (i.e. the first common ancestor) expressed as the number of hops to the root of the topological hierarchy considering only hypernymy relations. This measure is computed through a non-linear function.

The choice of a non-linear function derives from several considerations. The values of path length and depth, based on their definition, may range from 0 to infinity, while relatedness between two terms should be expressed as a number in the [0, 1] interval. In particular,

Table 4 Property Weights

Property	Weight
Antonymy	0.8
Attribute	0.7
Category Domain	1
Cause	0.6
Derived	0.8
Entailed by	0.7
Entailment	0.7
Hypernym	0.9
Hyponym	0.9
Member Holonym	0.5
Member Meronym	0.5
Member of Category Domain	1
Nominalization	0.7
Part Holonym	0.7
Part Meronym	0.7
Principle of	0.7
See Also	0.6
Similar To	0.5
Substance Holonym	0.5
Substance Meronym	0.5
Synonymy	1

when the path length decreases towards 0, the relatedness should monotonically increase towards 1, while it should monotonically decrease towards 0 when path length goes to infinity. We need a scaling effect with respect to the depth, because words in the upper levels of a semantic hierarchy express more general concepts than the words in a lower level. We use a non linear function for scaling down the contribution of subsumers in a upper level and scaling up those in a lower one.

The path length is calculated as follows (see (1)):

$$l(w_1, w_2) = min_j \sum_{i=1}^{h_j(w_1, w_2)} \frac{1}{\sigma_i} \qquad (1)$$

where l is the best path, w_j and w_i are the terms, $h_j(w_i, w_j)$ are to the number of hops of the j-th path and σ_i corresponds to the weight of the i-th arch of the j-th path. The *depth* of the subsumer of w_i and w_j is calculated by considering only the hypernymy relations and is defined as the number of hops from the subsumer to the root of the hierarchy. The Fig. 8 shows an example that illustrates the measure of the best path and depth.

Even if there is a path with only one hop, the path that cross through the node Z is chosen because the weighted sum returns a value of 2. Furthermore, evaluating the relations of hyponymy and hypernymy only the node S represents the subsumer whose depth is equal to 1 being the node R the root of the hierarchy.

Using the *best path* and *depth* between each pair of terms, the *Semantic Relatedness Grade* is calculated using (2) proposed in a previous authors' work [54]:

$$SRG(\upsilon) = \sum_{(w_i, w_j)} e^{-\alpha \cdot l(w_i, w_j)} \frac{e^{\beta \cdot d(w_i, w_j)} - e^{-\beta \cdot d(w_i, w_j)}}{e^{\beta \cdot d(w_i, w_j)} + e^{-\beta \cdot d(w_i, w_j)}} \qquad (2)$$

where (w_i, w_j) are pairs of terms in the υ document, $l(w_i, w_j)$ is the best path as shown in (1), $d(w_i, w_j)$ is the depth of the common ancestor of the two terms, while α and β are parameters whose values have been defined by experiments.

The WSD process computes the score for each sense of the considered term using the proposed metric. The best sense associated with a term is the one that maximizes the *SRG* obtained from the semantic relatedness among all the terms of the document. The variant of this process is based on the choice of a window of context reducing the set of analyzed terms and not to the whole document. For this purpose, the document is divided into grammatical periods, defined by punctuation such as the dot, question mark and exclamation mark. The *semantic relatedness* of a sense is calculated with respect to each sense of each word belonging to its own window and not to the whole document.

The pseudo-code of the algorithm used to determine the best sense of a term is shown in Algorithm 1.

Algorithm 1 Best Sense Algorithm.

 1: **procedure** BEST_SENSE(W_t)
 2:    **for** each sense $S_{t,i}$ of target word W_t **do**
 3:       set $Score\_St_{t,i} = 0$
 4:       **for** each word $W_j \neq W_t$ in windows of context **do**
 5:          init array $temp\_score$
 6:          **for** $each sense S_{j,k} of W_j$ **do**
 7:             $temp\_score[j] = SRG(S_{t,i}, S_{j,k})$
 8:          **end for**
 9:          $Score\_S_{t,i} = Score\_S_{t,i} + MAX(temp\_score)$
10:       **end for**
11:       **if** $best\_sense\_score < Score\_S_{t,i}$ **then**
12:          $best\_sense\_score = Score\_S_{t,i}$
13:          $best\_sense = S_{t,i}$
14:       **end if**
15:    **end for**
16:    return $best\_sense$
17: **end procedure**

The best sense of the given term is the one with the highest score obtained by the estimation of the semantic relationship with all the other terms of a given window of context. Both these approaches have been evaluated in the experimental section. The main advantage of the WSD algorithm is its ability to infer a correct semantic of a document, by deducing the correct meaning, or sense of words contained in the documents. This is extremely helpful in documents where keywords suffer the problem of polysemy. A possible limitation of the algorithm instead is the computational load needed to perform it, since the computation of such score for each pair in the collection of documents could be a potentially heavy-demanding task. However, it should be noted that such operation is typically
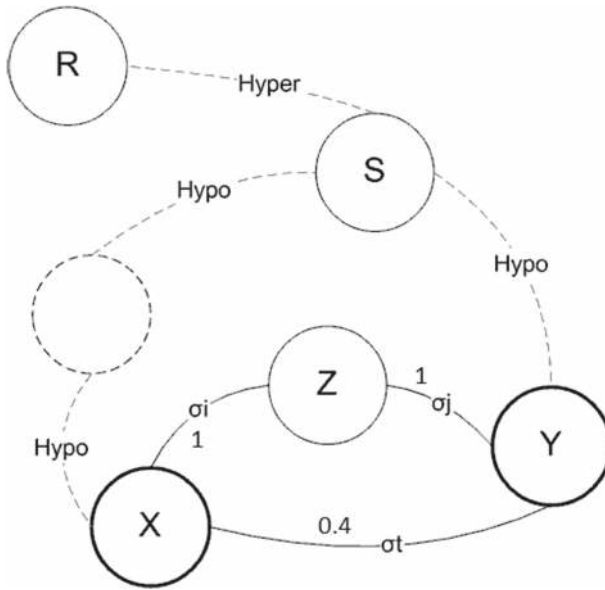
Fig. 8 Example of semantic network, best path and depth

executed off-line and doesn't have strict real-time requirements. Moreover, the variant pro-posed for the window of context looks straight into this direction, that is to speed up the algorithm making it more efficient while keeping intact its effectiveness.

3.8 Textual topic detection

The purpose of the *Textual Topic Detection* module is to recognize the main topic (i.e. *Topic Concept*) of the analyzed document represented by a concept in our knowledge base.

Figure 9 shows the module tasks.

The first step is the *tokenization* used to obtain the list of terms in the document. The terms are sent to the *Word Sense Disambiguation* module (see Section 3.7), to find the right meanings of the terms. For each of them the system generates a DSN using
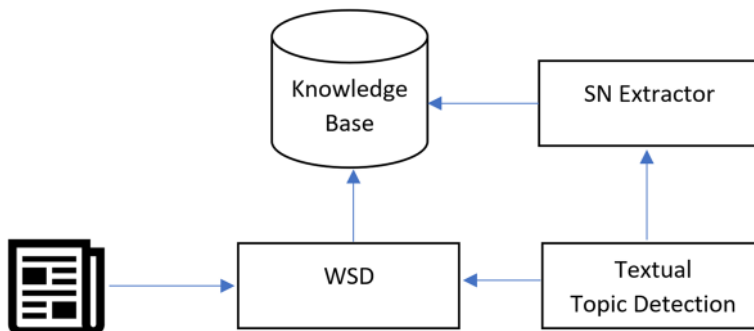

Fig. 9 Textual topic detection scheme

the *SN Extractor* (see Section 3.6) based on our knowledge base (see Section 3.4). The system builds a DSN from each recognized concept (seed concept) and it calculates the intersections between the generated semantic networks counting the number of common nodes. The common nodes represent the degree of representation of the considered concept (i.e. the corresponded DSN) with respect to the whole document. This measure is called *Sense Coverage*. The number of common concepts would facilitate more generic concepts because a large DSN will be generated from them. For this reason a scaling factor is used depending on the depth of the seed concept (depth), calculated as the number of hops to the root of WordNet considering only hypernymy relationships. Thus, the *Topic Concept* will be the concept having the best trade-off between *SenseCoverage* and *Depth*.

The used formula is shown in (3):

$$TopicConcept = max(Depth(C_i) * SenseCoverage(C_i)) \tag{3}$$

where C_i is the i-th concept returned from WSD. We consider only the *Noun* lexical category because we argue that it is more representative of the documents subject [13].

The pseudo-code of the implemented algorithm is shown in Algorithm 2.

Algorithm 2 Textual Topic Detection Algorithm.

1: **procedure** TOPICCONCEPT($ConceptList$)
2:      $BestConceptScore = 0$
3:      **for** each concept C_i in ConceptList (after WSD) **do**
4:          $ScoreC_i = 0$
5:          $DSN\_C_i = BuildDSN(C_i)$
6:          $CoverC_i = 0$
7:          **for** each concept $C_j \neq C_i$ in ConceptList **do**
8:              $DSN\_C_j = BuildDSN(C_j)$
9:              $NumberOfCommonConcept = Match(DSN\_C_i, DSN\_C_j)$
10:             $CoverC_i = CoverC_i + NumberOfCommonConcept$
11:         **end for**
12:         $ScoreC_i = depth(C_i) * CoverC_i$
13:         **if** $BestConceptScore < ScoreC_i$ **then**
14:             $BestConceptScore = ScoreC_i$
15:             $TopicConcept = C_i$
16:         **end if**
17:     **end for**
18:     return TopicConcept
19: **end procedure**

As previously explained, this algorithm benefits from the semantic network generated from the WSD algorithm to compute the coverage between a limited number of concepts, already well-defined from a semantic point of view. Moreover, the algorithm includes a trade-off factor which mitigate the problem of favoring more generic concepts in topic identification. As a consequence, the performance of this algorithm are bound to the goodness of the WSD algorithm. For this reason, we also tested its performance without the WSD step, as it will be shown when discussing the results in Section 4.

3.9 Visual topic detection

This module employs visual representations of a concept and it uses multimedia features extracted from them to recognize the document topic. Figure 10 depicts the main aspects of the process.

The images are taken from the document and sent to the *Visual Matcher* module, which estimates the similarity with respect to the knowledge base visual collections. The result of the matching is a list of concepts retrieved by the *Visual Topic Detection* module which, with an appropriate semantic analysis, computes for each of them a score based on *sense coverage*, *depth*, matching measure and frequency.

The Visual Matcher analyzes an image using a set of multimedia features extracted using a *Feature Extractor*. These features are compared with the ones in the system knowledge base through a distance metric which measures the proximity between two points belonging to the same metric space (i.e. the feature space).

Since every visual node in the database is linked to a concept, the result is a list of concepts, sorted by the evaluated metric. Three types of metrics have been used: *manhattan*, *euclidean* and *cosine*.

An image appears to be more specific than a textual content but it shares some characteristics with different concepts from a visual features point of view. The *Visual Matcher* is in charge of assigning the right meaning to a visual representation, as well as the *Word Sense Disambiguation* with text. On the other hand, the *Visual Topic Detection* is used to increase the accuracy of the results. It performs a semantic analysis starting from the list of the most relevant concepts returned by the matching phase as for the *Textual Topic Detection*.

For each concept returned from the matching step, we compute a score and, as in the case of text, the *TopicConcept* is determined by the largest sense coverage. The score is computed with a formula that represents an extension of the one presented in the textual case. The used parameters are listed below:

– *Sense Coverage*: semantic analysis between concepts. The number of common nodes between concepts is counted starting from the extraction of their subnets with the help of the *SN Extractor* module, as in the textual case;
– *Depth*: the specialization of the concept calculated as the number of hops to reach the root node of the WordNet hierarchy;
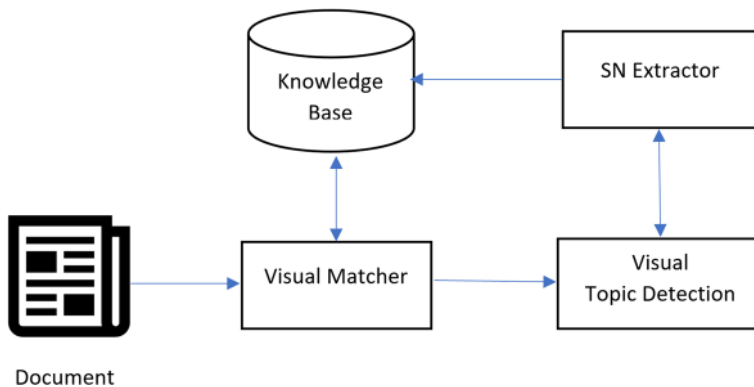

Fig. 10 Visual Topic Detection component

- *Frequency*: given the multiple visual representations for each concept, the matcher can return the same concepts in the ordered classification. These concepts are merged, increasing their *frequency*, calculated as the number of times that the same meaning is present in the ranking;
- *Distance*: the value included in [0,1] (0 indicates that they are the same, vice versa 1), represents the similarity between two images represented in the same metric space, returned by the matching phase. If a concept is present several times the average will be assigned as score.

These parameters are calculated for each concept and the score is expressed according to (4):

$$TopicConcept = max((1 - Distance) \cdot SenseCoverage \cdot Depth \cdot Frequency) \quad (4)$$

The role of this component is to smooth possible mistakes in the matching phase. The main topic assigned to images in a document will correspond to the concept that will have a higher degree of relatedness from both semantic and visual properties point of views.

The textual and visual classifiers have been combined using an ad-hoc module. Figure 11 shows a functional scheme for the combining procedure.

The score of each concept calculated during the textual and visual topic detection processes is normalized with respect to returned the best score, and it is normalized in the [0,1] interval. The combination of textual and visual classifications can be performed according to various schemes. In our experiments we use *SUM* and *OWA* operators [73].

3.10 Semantic tag cloud summary

The proposed approach based on Semantic Tag Clouds differs from the normal tag cloud in the evaluation of the relevance between terms taking into account meanings (concepts). For its implementation we need to define a metric that combines statistical and semantic properties of terms in a document, whose value is used to resize the font of the terms. The proposed equation is based on two parameters: the *Syntactic-Semantic Grade* (*SSG*) and the *Semantic Relatedness Grade* (*SRG*). The *SRG*, described in Section 3.7, indicates the degree of semantic relatedness between a term and the rest of a document. The *SSG* defines the relevance of a term in a document combining statistical and semantic information. The metric proposed in [66] has been improved adding the contribution of semantics properties as the degree of polysemy that can be considered as a measure of the ambiguity in the use of a term if it can assume several senses.

Thus we define the reciprocal of the polysemic degree ω_i of the term i as:

$$\omega_i = \frac{1}{poly_i} \quad (5)$$

poly(i) being the polysemy (number of senses) of i.

As an example, the word *dog* has seven senses in our knowledge base. Assuming that all meanings have equal importance and occurrence, the probability that this word is used to express a specific meaning is equal to 1/7.

The used formula is expressed in the following (6):

$$SSG_{i,k} = \frac{(0.5 + 0.5(TF_{i,k}/TF_{max,k})) \cdot \omega_i}{\sqrt{\sum_{i \in k}(0.5 + 0.5(TF_{i,k}/TF_{max,k}))^2(\omega_i)^2}} \quad (6)$$
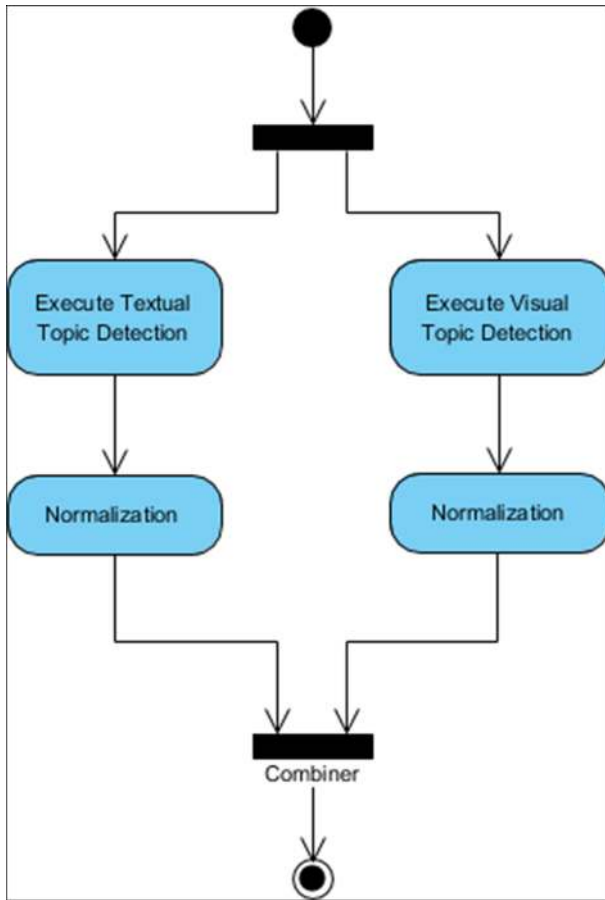
Fig. 11 Combiner

where k is a generic document, i is the i-ith term of the k document, $TF_{i,k}$ is the *Term-Frequency*, the number of times the term i is present in the document k; $TF_{max,k}$: the maximum frequency present in the document k; ω_i: degree of polysemy of the term. More concepts are associated with a term the higher the degree of polysemy;

As we can see, the value of this factor is in the interval [0,1] but its denominator is similar to the sum of the SSGs of all the terms, hence its score will return values closer to 0 than to 1. The *SRG*, on the other hand, assumes theoretical values in the range $[0, \infty]$. Therefore, we have to normalize these parameters, as shoen in (7):

$$norm = \frac{P_{i,k} - min_k}{max_k - min_k} \tag{7}$$

where P represents either the *SSG* or *SRG* parameter, while i and k have the usual meaning as previously expressed. This normalization was chosen because we are interested in the measure of the difference of relevance between document terms in the same interval in order to compare the two parameters on the same scale.

At this point we are able to define in (8), the *Global Score* ($GS_{i,k}$) of a term i in the document k as:

$$GS_{i,k} = \frac{SSG_{i,k} + SRG_{i,k}}{2} \tag{8}$$

Since the result of the sum can belong to the interval [0,2], it is divided by its maximum value to have a normalization in the same interval of other parameters. The term with the best overall score represents the one that contains the best combination of statistical and semantic information in the context in which it is presented. Once all the *GSs* are calculated for each term of the document it will be possible to generate a *Semantic Tag Cloud Summary*. The knowledge base is further used to add to the analyzed terms their synonyms to enrich the informative content of the Semantic Tag Cloud. Moreover, the tag clouds have been extended representing a concept with a visual representation.

3.11 Semantic tag cloud category

Semantic tag clouds are also used to synthesize and display a list of documents clustered around the common main topics retrieved from the fetching phase. In this case, the font size indicates the number of documents in the collection belonging to a topic. Figure 12 shows the involved modules

Given a user query, the fetcher retrieves a list of relevant documents. For each of them, the topic detection task is performed to determine the main topic. The number of documents with the same topic (i.e. frequency) is calculated and it is used to set the font size of the word representing the concept. The purpose of using this kind of topic categorization and visualization through semantic tag clouds is to help the user in the browsing process.

3.12 Use case example

The proposed example simulates the behavior of a user while browsing the information content provided by the system. We want to show a real use case example based on the proposed algorithms, interface and knowledge base. The data set is limited to documents belonging to the animal domain.

In this scenario, the user is interested in the retrieval of general information about dogs. With this regard, he/she submits a query by typing the term *"dog"*. This query type is known as *single word* query. The query is passed to the fetcher which retrieves all documents where the term dog is present. A Semantic Tag Cloud Category is generated from the list of documents retrieved using the topic detection task to classify each document with the related topic concept. The result is shown in Fig. 13.

As explained in Section 3 about the STCC module, the font size indicates the number of documents belonging to a certain category. The tag clouds based on a relation to the terms with semantic meanings is extended with a visualization by images. In this scenario, the user, interested in knowing the domain of "dogs", may not be able to understand the meaning from the textual view only given the names of the breeds often not immediately related to a specific dog. In fact, he/she learn the true concept viewing the glossary, reports and other multimedia representations. Starting from the Semantic Tag Cloud Category it will acquire a partial knowledge of the various categories associated to the typed query. In Fig. 14 some simulated actions of the user during his research process are shown.

In [68] it was observed that creating a tag-based search system means that the processes of scanning and clicking on tags are easier than new query formulations. The user, once

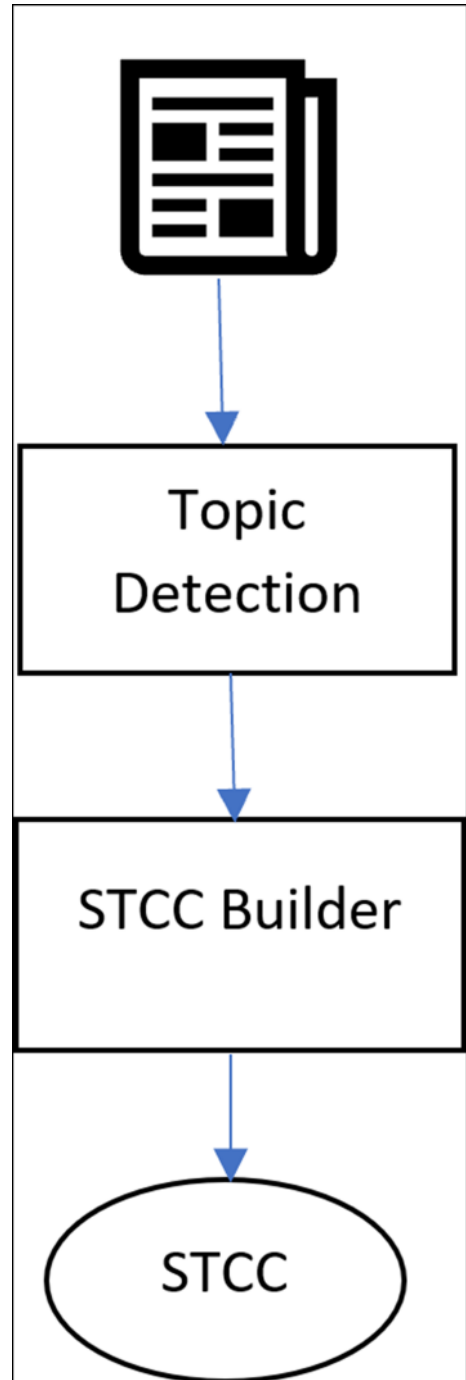**Fig. 12** Components for
Semantic Tag Cloud Category

Fig. 13 Semantic tag Cloud Category example

acquired general information from the categories, can simply select the domain of inter-est rather than writing a new query, following the conceptual process of his/her navigation. Assuming that the user is interested in a particular breed of dogs, for example to *"Spitz"*, he/she can click on the related tag to view the Semantic Tag Cloud Summary of all the doc-uments belonging to that category. The STCS operations are the same of those of the STCC except for the *left-click* event in which the original document will be displayed. Figure 15 shows the result of an STCS of one of these documents, the *Keeshond*, a sub-branch belong-ing to the spitz breed, and the operations that can be performed on the tag clouds. In addition, the original Wikipedia document is displayed.

In our tag clouds representation, the size of a term related to a concept represents its semantic and statistical relevance in the document compared to the other terms. In this exam-ple, the main features provided by the system are shown. In fact, we can see the basic role of the lexicon-semantic knowledge base. It is a fundamental element for textual and visual document classification, subnets generation and additional semantic information retrieval. Starting from a concept, all its topological paths are obtained with a simple query, thanks to which it is possible to categorize the documents at different levels of granularity. More-over, using our semantic network, a concept can be expressed through multiple multimedia representations.

4 Experimental results

In this section we discuss the results of several experiments carried out to evaluate the performances of our framework. We have tested our system using well defined data to represent a real use of the proposed algorithm and metrics. Furthermore, to allow a compar-ison between different approaches, the performance of the developed classifiers have been
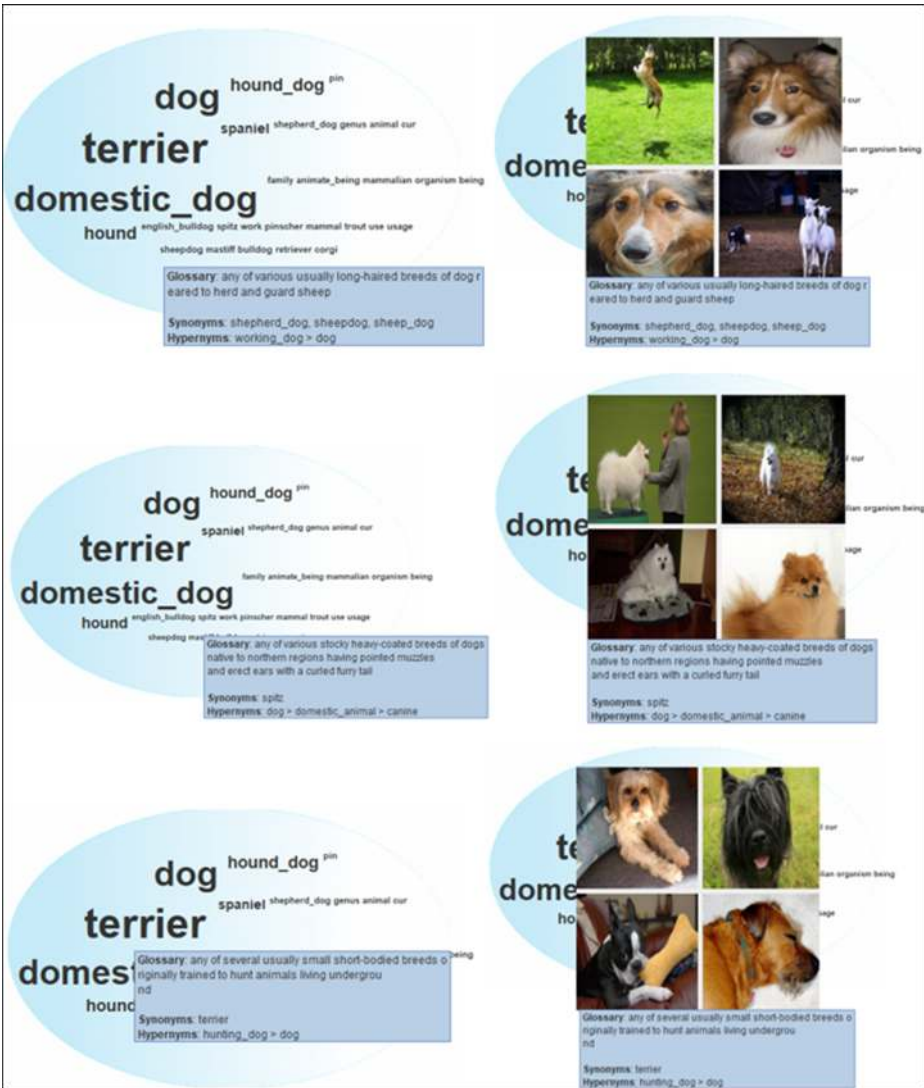
Fig. 14 Interactions with STCC

evaluated by appropriate standard measures. We perform our experiments on the following system components:

– Textual Topic Detection;
– Visual Matcher;
– Visual Topic Detection;
– Combined Topic Detection.

To evaluate the accuracy of the system outputs, the outputs must be compared with a ground truth. We choose to carry out the evaluation task using a test set of previously cata-loged documents to perform a comparison with the results by our system using the proposed

Fig. 15 Interactions with STCS

approach. To accomplish this goal, documents in the ground truth should have multimedia contents, in particular text and images. In this way, it will be possible to perform tests for all the components listed above. Hence, to build such a collection, we fetched documents from *Wikipedia*. In order to associate the document topic with the concept of the ontology, we query the knowledge sources and extract the necessary contents. We extract the

DSN identified by a concept of the knowledge base. For each concept we query *BabelNet* [51], a knowledge base that integrates more than 20 semantic networks, including *WordNet*, *WikiPedia* and *DBPedia*. BabelNet has been used to preserve the consistency of the concepts in the proposed knowledge base through the WordNet ID. In this way we ensure that the topic of downloaded documents is related to the corresponding WordNet synset. Given the WordNet ID, the *BabelSynset* is retrieved together with a field for each integrated external source. In particular, the ID of Wikipedia document. Using this ID we can download the multimedia document related to the concept. We use a SPARQL [5] query to access to DBPedia information directly connected to Wikipedia and retrieve the textual component. In other words, BabelNet is used as a semantic bridge between WordNet and contents retrieved from external sources to automatically build our test set in an automatic way, without human experts or manual construction of labeling documents. Moreover, images are fetched from the Wikipedia html document through the *Jsoup* library [32].

The result of this algorithm is a list of multimedia documents whose structure is composed of the following fields:

–   *Title*: title of the document;
–   *Body*: text of the document;
–   *Images*: list of images associated to the document (three images have been downloaded for each document);
–   *ActualConcept*: the main topic linked to the implemented knowledge base.

The animal domain has been chosen because it is the most populated conceptual domain both from a textual and visual point of view. It is represented by the concept *"animal"* in the knowledge base. Each single document with its structure is stored in the file system in JSON format.

Table 5 shows general statistics about the test set.

The aim of our results analysis is twofold and it is obtained by two strategies.

In the first strategy our goal is to visualize the ability of the topic detection to proper classify a document. Since the topic detection is an algorithm mainly used for the categorization of documents through the *Semantic Tag Cloud Category* component, the prediction of a more generic concept belonging to the hierarchy of the expected one has also been considered correct. It has been calculated through a topological analysis of the related path, in particular if it presents only hypernymy relationships. In this way it is possible to calculate the *accuracy* as the number of correct predictions.

In the second strategy we want to perform a qualitative analysis of the classifier. In other words, we want to measure the accuracy in terms of hierarchical proximity between the predicted and the expected concept. For example, if a document about *"beagle"* is classified

Table 5 Test set statistics

Field	Value
Num. Documents	1102
Avg length Doc.	754.15
Max length Doc.	1072
Min length Doc.	353
Num. Images	3306
Avg Images	3

as *dog*, it would be a correct prediction but certainly less specific than beagle.Therefore, the semantic similarity is calculated among the correct evaluated classifications considering only the hypernymy relation **to assess how far is the predicted topic from the actual one**.

The testing process has been completely automatized and developed with Java procedures with the support of Neo4J library for querying the knowledge base using Cypher language. All documents in the dataset have been analyzed by the classifier to have the related topic concept; it is compared with the expected topic and the two measurements listed above are evaluated.

The metrics have been implemented through the Cypher query, shown below, which extracts from the ontology the hierarchical path consisting only of hypernymies between the classification returned by the classifier and the expected concept in the ground truth.

```
MATCH
path=shortestPath((actual :CONCEPT)- [ :HYPERNYM*]
- >(predicted :CONCEPT))
WHERE
actual .IdConcept = {i dActual} ANO
predicted .IdConcept = {idPredicted}
Return path
```

Figure 16 shows some examples to better understand our strategies.

We assume to classify a document that has as main topic the concept *beagle*.

Three examples are presented. For each of them, the classified concept is drawn in orange while the expected concept in green. In the case (A) the query will return an existing path between the *dog* and *beagle* concepts. Since the path is a part of the semantic network about the *animal* domain, the prediction is correct and the semantic distance is calculated in terms of the number of hops between the two concepts.

In the case (B), a hypernymy path is found between *organism* and *beagle* so it should be evaluated as correct, but since it is a more abstract subject than *animal*, it is considered wrong. This is shown by the *animal* node in the path returned by the query. In the case (C), even if the *tiger* concept is belonging to the *animal* domain, the classification is wrong because the path will consist of both hypernymies and hyponymies, in fact the query will not return any result.

The test process consists in evaluating the two strategies according to the methodology before explained for each document in the test set. The *animal* node has been used as conceptual boundary. The second strategy is performed to assess the topic recognition if a classifier tends to predict generic or specific concepts in terms of average number of hops.

4.1 Visual matcher testing

The test of this component is able to measure the performance of metrics and features used to represent an image. The measures are performed on each image in the test set using 3306 examples. Figure 17 shows the accuracy computed on the test set.

The experimental results show that the *JCD* feature with the *cosine similarity* metric is the more accurate than the other combinations, immediately after we have *CEDD* with the *euclidean* distance. In general the visual matching do not show high accuracy. We argue that the reasons are:

1. hardware restrictions: it was not possible to analyze all images from *ImageNet* at this stage of our research; the number of images have been limited at 15 for each concept;
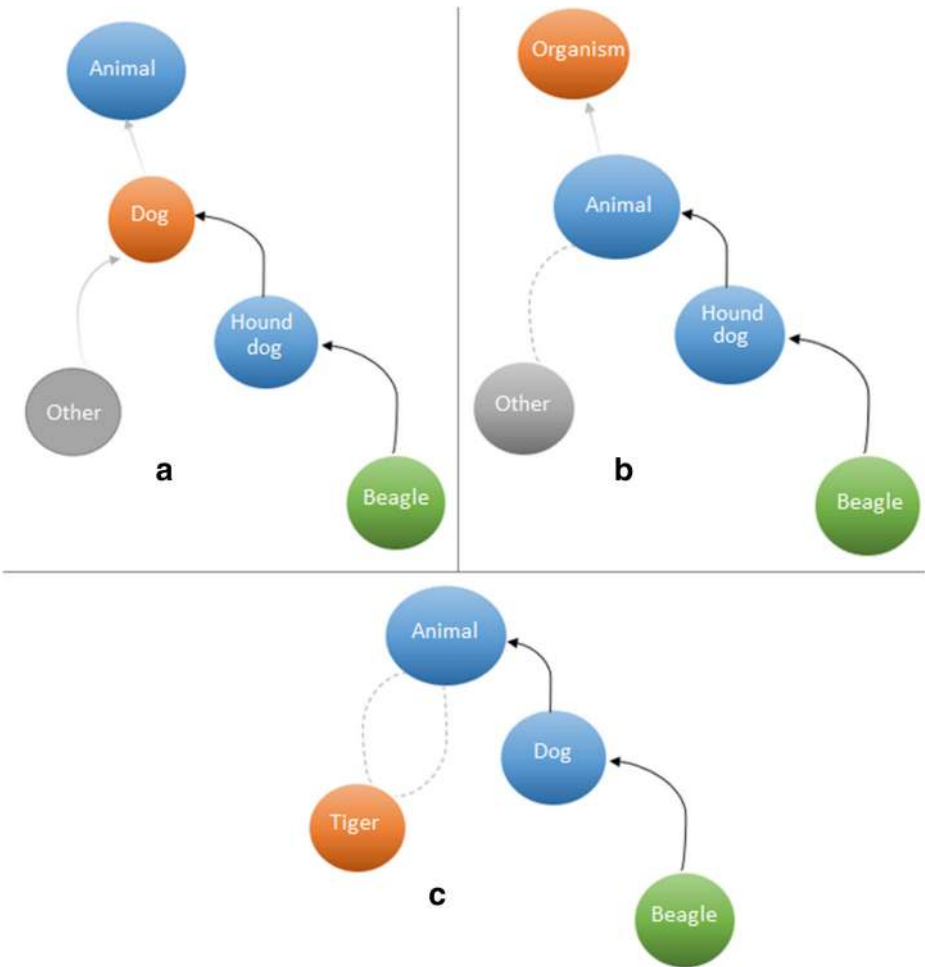
Fig. 16 Examples of classification of a document

2. quality of visual representations: several images in *ImageNet* do not represent in a clear way the associated concept;
3. global features: the use of global features, combined with the quality of some images fails to obtain good results.

From a technical point of view, such a behavior is natural and predictable. In particular, considering the third reason above discussed. In fact, global features are compact descriptors, i.e. vectors of numbers, extracted from images considering them as a whole and hence considering also noising elements, such as background of the image, etc. On large data collections with a high number of classes, global features don't have a high discriminating power. However they have the advantage of being fast for computing distances because of their compactness, which is an important factor to consider in large databases.
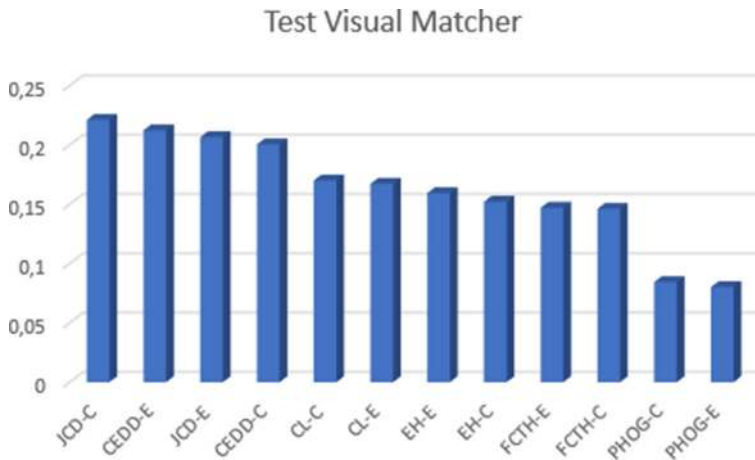
The results are shown in Table 6.

Test Visual Matcher

Fig. 17 Accuracy of visual matcher

The best combinations of feature and metric, i.e. JCD-C, CEDD-E and JCD-E, have been chosen to test the visual topic detection algorithm.

4.2 Test topic detection

This section describes the results of textual, visual and combined topic detection tasks. The aim is to evaluate the differences between the respective modules and to show that the combination of textual and visual analysis gives better performance than the single classifiers. The total number of documents used in our experiments, as reported in Table 5, is 1102.

4.2.1 Visual topic detection

In our approach, given the list of concepts obtained during the matching phase, a semantic analysis is performed between them. If a document has more than one image, a *merging*

Table 6 Summary table of Visual Matcher tests

Combination	Accuracy	Num.Correct	Average Path
JCD-C	0,2211	731	0,45
CEDD-E	0,2123	702	0,48
JCD-E	0,2065	683	0,44
CEDD-C	0,2005	663	0,43
CL-C	0,1699	562	0,32
CL-E	0,1672	553	0,33
EH-E	0,1594	527	0,55
EH-C	0,1521	503	0,52
FCTH-E	0,147	486	0,39
FCTH-C	0,146	483	0,33
PHOG-C	0,0843	279	1,09
PHOG-E	0,0801	265	1,06

Fig. 18 Accuracy of visual topic detection
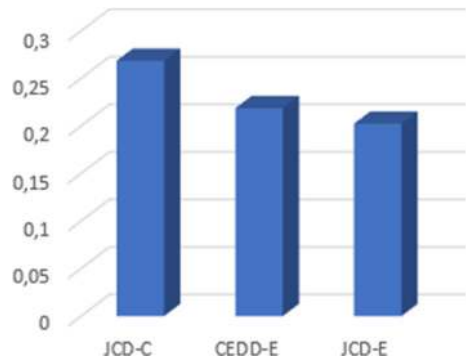
operation of individual results is performed before executing the semantic analysis. If the same concept is present in two different lists, the two instances are joined together, increasing frequency and averaging their scores. In this way, a single concept is predicted for the visual contents of a document.

Three different visual topic detection tests have been carried out with different features and metrics. The combinations that achieved better accuracy from the visual matcher test in Section 4.1 have been selected and the results are shown in Fig. 18.

It is interesting to notice that the results are close among different combinations of descriptors and distance metrics and they follow the ones obtained in the matching phase. In particular the feature *JCD* with the *cosine similarity* metric gives the best accuracy as also shown in Table 7.

It is also worthy to notice another trend from results in Table 7. There is an increase in the average path for all the combinations. This is due to the topic detection algorithm which tends to generalize the concept classification.

4.2.2 Textual topic detection

We test textual topic detection using two scaling strategies of the document dimension. The first is to calculate the *SRG* of a sense related to a term by semantically comparing it with all the terms of the whole document. The second one uses a window of context (WOC). In this case the used window corresponds to a grammatical period. The end of a period is identified by punctuation marks.

In Fig. 19 the accuracy of the two strategies are shown.

The *textual WOC topic detection* has a better accuracy and the results are in Table 8. As discussed in Section 3.7, the main purpose of using of a window of context is to increase the speed of execution of the algorithm lightening its computational load. The results shown in

Table 7 Summary table of visual topic detection test

Combination	Accuracy	Num.Correct	Average Path
JCD-C	0,2686	296	1,02
CEDD-E	0,2186	241	0,86
JCD-E	0,2023	223	1,11

Fig. 19 Accuracy of textual topic detection

Fig. 19 and Table 8 also prove that we got a secondary beneficial effect, that is the improvement of accuracy in textual topic detection. Such a behavior was also unsurprising, since we expect words in sentences to follow a sort of principle of locality, i.e. words semantically closer also appear closer in text.

Our textual topic detection algorithm has been compared with standard baselines presented in literature to put in evidence its performances. A robust analysis of these results allow us to have a more accurate and reliable measure of the effectiveness of our framework. We use for our purposes two standard algorithms widely used in document analysis: *LSA-Latent Semantic Analysis* [41] and *LDA-Latent Dirichlet Allocation* [6] used for textual topic detection of documents. LSA is a document vector representation technique based on the bag-of-words model. The method use the SVD (singular value decomposition) approach in order to identify relational models between words and concepts contained in a document. LSA is based, like other statistical topic-modeling models and algorithms, on the principle that terms used in the same context tend to have similar meanings. LDA is a model for text mining based on statistical approach. The basic idea is that each document can be modeled as a mix of topics and each topic represents a discrete probability distribution of the terms that compose it. As for LSA, also in LDA the document follows a bag-of-words approach.

The results obtained from experiments are in Table 9.

The same results are shown in Fig. 20.

The results show that our approach outperforms both LSA and LDA. In fact, our approach exhibits an accuracy of 0,55 where LSA and LDA only achieve an accuracy of 0,29 and 0,37 respectively. LSA shows good performances in the average path. This last consideration is very important to understand the nature of our approach since it shows the conceptual difference with other algorithms. In fact, our approach exploits semantics information to better understand the semantic context of document topics, allowing more general concept classification but still correct; in this context other algorithms fail. On the other hand, it should be also considered that the value of accuracy for LSA and LDA is affected by

Table 8 Summary table of textual topic detection test

Combination	Accuracy	Num.Correct	Average Path
Textual TD WOC	0,5544	611	1,84
Textual TD	0,4927	543	1,95

Table 9 Comparison of textual topic detection methods

Algorithm	Accuracy	Num.Correct	Average Path
LSA	0,294	324	**1,48**
LDA	0,37	407	1,66
Our Textual TD WOC	**0,5544**	**611**	1,84

the impossibility of associating some topics generated by the model with the corresponding WordNet concepts. In our vision this is a hard drawback because they neither give a good level of formalization nor a sufficient degree of generalization.

Moreover, we argue that LSA can only partially capture polysemy (i.e., multiple meanings of a word) because each occurrence of a word has the same meaning due to the word being represented as a single point in space. The behavior results in the vector representation being an average of all the different meanings of a term in the corpus, which can make it difficult for comparison [41]. Other limitations is in the bag-of-words technique, where a text is represented as an unordered collection of words.

The experimental results show a clear difference between text and visual classifiers and their evaluation is used to choose the weights assigned in the combination function. On the other hand, textual analysis has a higher, and therefore worse, *average path* than the visual one, putting in evidence that images represent a more specific way to represent a concept than textual content.

4.2.3 Combined topic detection

In the previous subsections we provide comparisons and explanations for each single component of our topic detection approach providing results to show the improvements of our algorithms. Now we are in the position to test the use of combined multimedia information and have a reliable evaluation of possible strategies.

The combination features used in our work have been described in section 3. Moreover, it is possible to define some schemes in which the visual and textual classifiers can be classified to improve the whole system performance. In particular, it is possible to exploit
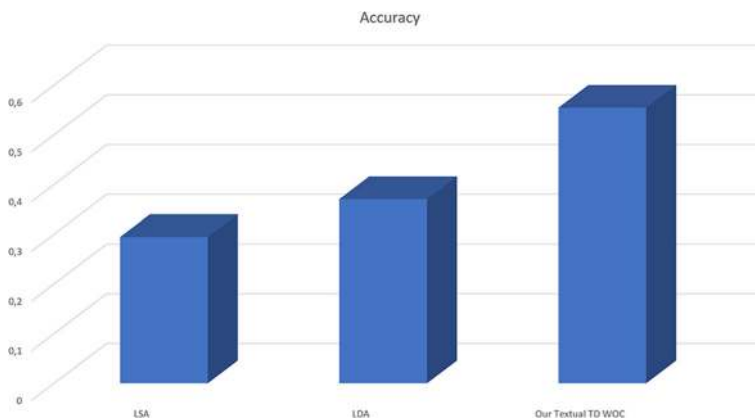

Fig. 20 Comparison of textual topic detection algorithms

Table 10 Weights for the SUM combinations

Combination Name	Textual weight	Visual weight
A	0.55	0.45
B	0.7	0.3
C	0.5	0.5

the text content to restrict the domain of the visual matching and evaluate the multimedia features only among the visual representations of the concepts in the textual document. The following schemes have been defined and tested:

↪ *After WSD*: after the *WSD* process execution, all the concepts associated with each term in the document are used as domain for visual matching. In this scheme, visual topic detection depends not only by the textual content but also by the performance of the *Word Sense Disambiguation* module;

↪ *Before WSD*: after the tokenization phase and before the execution of the *WSD* process, the set of all senses of each term contained in the document is used as a visual domain for matching. In this scheme the visual topic detection is dependent by the text content but independent by the the WSD module;

↪ *Normal*: matching is applied to the whole domain without any restrictions. Textual and visual topic detection are completely independent.

The techniques used to combine our classifiers are *SUM* and *OWA* operators. Each proposed scheme has been evaluated with various combinations of weights associated with textual and visual classifiers, except for OWA operator using a fuzzy logic approach. Table 10 shows the used weights.

Combination A is used in the tests previously shown in which the textual topic detection showed better performance; with the combination B we want to increase the difference in the trust between classifiers. The combination C is a balance between the scores of the two classifiers. Furthermore, a combination with the OWA operators (scheme indicated with the letter D in Fig. 21) has been tested using the weight vector $\vec{w} = [0.35, 0.65]$. We perform a total of 12 types of combinations for our tests.
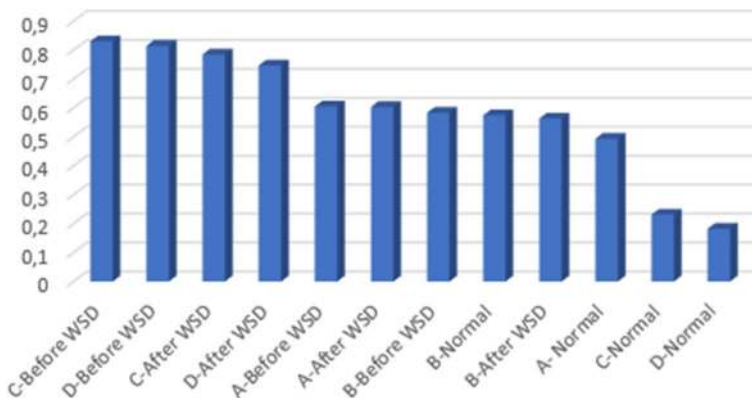

Fig. 21 Accuracy of combined topic detection

Table 11 Summary table of combined topic detection test

Combination	Accuracy	Num.Correct	Average Path
C-Before WSD	0,83030853	915	1,33
D-Before WSD	0,81488203	898	1,28
C-After WSD	0,78493648	865	1,36
D-After WSD	0,74682396	823	1,34
A-Before WSD	0,60526316	667	1,61
A-After WSD	0,60344828	665	1,56
B-Before WSD	0,58352087	643	1,57
B-Normal	0,57441016	633	1,66
B-After WSD	0,56352087	621	1,54
A-Normal	0,49364791	544	1,66
C-Normal	0,23139746	255	1,15
D-Normal	0,18239564	201	1,17

The schemas without restrictions have a low accuracy. In particular, the normal schemas present a very low accuracy for the C and D combinations that correspond to the balanced weights of the fuzzy logic values. This is confirmed by the performance found in the previous tests on visual topic detection. The normal schema B has an accuracy comparable with the textual one in which there is a significant difference between the weight values.

The best results are given by the *Before WSD* and *After WSD* schemas. Table 11 shows the results values.

The *Before WSD* schemas with balanced weights among the classifiers are the ones with greatest accuracy because the restriction of the domain of visual matching only on the senses contained in a document improve the performance of visual topic detection.

A comparison of the best combined, textual and visual topic detections strategies is shown in Table 12.

The combination of textual and visual classifier shows excellent results compared with the single ones. Moreover, the combined classifier provides an improved quality of results accuracy also in terms of *Average Path*. This is due to the contribution of visual analysis in a restricted domain of the textual document content. In general, the experiments have shown that visual topic detection should not be used alone, but it can give a very useful contribution in combination with the textual topic detection due its intrinsic precision it the concept representation (i.e. low polysemy). Some results on the use of our approach from

Table 12 Summary table of textual topic detection test

Combination	Accuracy	Num.Correct	Average Path
C-Before WSD	0,83030853	915	1,33
D-Before WSD	0,81488203	898	1,28
Textual TD WOC	0,55444646	611	1,84
Visual TD JCD-C	0,26660254	296	1,02

a user point of view is discussed in [59] where empirical experiments to measure the user satisfaction have been presented.

5 Conclusion and future work

The use of synthetic representations for retrieved documents to satisfy user information needs is a basic tool in information retrieval processes to provide a solution to the information overload. In this context, the combination of formal knowledge representations, semantic analysis tasks and document visualization tools offer a concise view of contents and help people to find and understand information from original documents in an easy way.

In this paper a new strategy based on Visual Semantic Tag Cloud is proposed to visualize and browse categories and summaries generated by multimedia documents.

A fully-modular system architecture is proposed, covering all aspects typical of information retrieval systems. Novel metrics and algorithms to categorize information have been proposed based on textual and visual analysis.

A real use case to show the use of the system has also been proposed and discussed. Furthermore, the summary generation task uses a combination of semantic and statistical analysis. In general, Visual Semantic Tag Cloud offers a quick and accurate way to render document contents giving an overview of large amount of data with less cognitive effort.

Extensive experiments have been carried out to test several modules of the proposed system both on quantitative and qualitative dimensions, considering effective metrics for this kind of evaluation. Results for visual matching through global features show that there is the possibility for large improvements for this task considering different kind of features. The topic detection task has been also evaluated in all its aspects. First single testing of visual topic detection and textual topic detection has been carried out to assess them separately. The latter has been also compared with different classification schemes and state-of-the-art techniques, such as LSA and LDA algorithms. The obtained results have been very encouraging, showing better performances of our algorithms and techniques with respect to other state of art algorithms. Further experiments have shown that our proposed combined approach presents more important improvements both for a general accuracy and for the specificity of the classification. The tests have been performed on a real domain fully populated from a conceptual point of view a represented in different formats (i.g text and images).

In addition, a semantic network representation has been proposed based on a formal multimedia model in which the concepts are represented in textual and visual form. In this way, it is possible to implement the proposed semantic analysis algorithms and retrieve additional information with multimedia representations.

Given the observation made about visual results obtained using global descriptors, future works will be focused on the use of other multimedia features based on local descriptors, e.g. SIFT, SURF, SIMPLE and features extracted from activation layers of deep neural network architectures for image classification. In this context other application domains as semantic object recognition will be investigated using also other kinds of devices [63] together with the improvement of our knowledge base with specific conceptual domains [12, 61]. Moreover, we are interested in extending our knowledge base with other data formats such as audio data and implement smart human-computer interfaces. Audio features would play the role of further "sign" for representing the knowledge of abstract concepts. The inclusion of such novel kind of data would allow the analysis of other kind of multimedia-rich documents for multimodal retrieval.

References

1. Adrian A, Richard AD, Ann FK, Robert HM (2001) Linguistics: An introduction to language and communication. United States: Massachusetts Institute of Technology
2. Alguliev RM, Aliguliyev RM (2005) Effective summarization method of text documents. In: Web Intelligence, 2005. Proceedings. The 2005 IEEE/WIC/ACM International Conference on, pp 264–271, IEEE
3. Alguliev RM, Aliguliyev RM, Hajirahimova MS, Mehdiyev CA (2011) Mcmr: Maximum coverage and minimum redundant text summarization model. Expert Syst Appl 38(12):14514–14522
4. Begelman G, Keller P, Smadja F et al (2006) Automated tag clustering: Improving search and exploration in the tag space. In: Collaborative Web Tagging Workshop at WWW2006, Edinburgh, Scotland, pp 15–33
5. Bizer C, Schultz A (2009) The berlin sparql benchmark. International Journal on Semantic Web and Information Systems (IJSWIS) 5(2):1–24
6. Blei DM, Ng AY, Jordan MI (2003) Latent dirichlet allocation. Journal of machine Learning research 3(Jan):993–1022
7. Bohne T, Rönnau S, Borghoff UM (2011) Efficient keyword extraction for meaningful document perception. In: Proceedings of the 11th ACM symposium on Document engineering, pp 185–194, ACM
8. Bosch A, Zisserman A, Munoz X (2007) Representing shape with a spatial pyramid kernel. In: Proceedings of the 6th ACM international conference on Image and video retrieval, pp 401–408, ACM
9. Caldarola EG, Picariello A, Rinaldi AM (2015) Big graph-based data visualization experiences: The wordnet case study. In: 2015 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K), 1, pp 104–115, IEEE
10. Caldarola EG, Rinaldi AM (2015) Big data: A survey: The new paradigms, methodologies and tools. In: DATA 2015 - 4th International Conference on Data Management Technologies and Applications, Proceedings, pp 362–370
11. Caldarola EG, Rinaldi AM (2017) Big data visualization tools: A survey: The new paradigms, methodologies and tools for large data sets visualization. In: DATA 2017 - Proceedings of the 6th International Conference on Data Science, Technology and Applications, pp 296–305
12. Caldarola EG, Rinaldi AM (2018) A multi-strategy approach for ontology reuse through matching and integration techniques. Advances in Intelligent Systems and Computing 561:63–90
13. Caldarola EG, Rinaldi AM (2016) Improving the visualization of wordnet large lexical database through semantic tag clouds. In: Big Data (BigData Congress), 2016 IEEE International Congress on, pp 34–41, IEEE
14. Castano S, Ferrara A, Montanelli S (2003) H-match: an algorithm for dynamically matching ontologies in peer-based systems. In: Proceedings of the First International Conference on Semantic Web and Databases, pp 218–237, Citeseer
15. Cattuto C, Schmitz C, Baldassarri A, Servedio VitoDP, Loreto V, Hotho A, Grahl M, Stumme G (2007) Network properties of folksonomies. Ai Communications 20(4):245–262
16. Chang Y-C, Hsieh Y-L, Chen C-C, Hsu W-L (2017) A semantic frame-based intelligent agent for topic detection. Soft Comput 21(2):391–401
17. Chatzichristofis S, Boutalis Y, Lux M (2009) Selection of the proper compact composite descriptor for improving content based image retrieval. In: Proc. of the 6th IASTED International Conference, 134643, p 064

18. Chatzichristofis SA, Boutalis YS (2008) Cedd: color and edge directivity descriptor: a compact descriptor for image indexing and retrieval. In: International Conference on Computer Vision Systems, pp 312–322, Springer
19. Chatzichristofis SA, Boutalis YS (2008) Fcth: Fuzzy color and texture histogram-a low level feature for accurate image retrieval. In: Image Analysis for Multimedia Interactive Services, 2008. WIAMIS'08. Ninth International Workshop on, pp 191–196, IEEE
20. Chen Q, Guo X, Bai H (2017) Semantic-based topic detection using markov decision processes. Neurocomputing 242:40–50
21. Chen Y-X, Santamaría R, Butz A, Therón R (2009) Tagclusters: Semantic aggregation of collaborative tags beyond tagclouds. In: International Symposium on Smart Graphics, pp 56–67, Springer
22. Danesi M, Perron P (1999) Analyzing cultures: An introduction and handbook. Indiana University Press
23. Deng J, Dong W, Socher R, Li L-J, Li K, Fei-Fei L (2009) ImageNet: A Large-Scale Hierarchical Image Database. In: CVPR09, pp 1082–9873
24. Fujimura K, Fujimura S, Matsubayashi T, Yamada T, Okuda H (2008) Topigraphy: visualization for large-scale sparse tag clouds. In: Proceedings of the 17th international conference on World Wide Web, pp 1087–1088, ACM
25. Gemmell J, Shepitsen A, Mobasher B, Burke R (2008) Personalizing navigation in folksonomies using hierarchical tag clustering. In: International Conference on Data Warehousing and Knowledge Discovery, pp 196–205, Springer
26. Hammond T, Hannay T, Lund B, Scott J (2005) Social bookmarking tools (i) a general review. D-lib Magazine, 2(4) 1082–9873
27. Haq MU, Shahzad A, Mahmood Z, Shah AA, Muhammad N, Akram T (2019) Boosting the face recognition performance of ensemble based lda for pose, non-uniform illuminations, and low-resolution images. TIIS 13(6):3144–3164
28. Harkovchuk AE, Korzun D (2019) Semantic information search service by person's face photo. In: Proceedings of the 24th Conference of Open Innovations Association FRUCT, pp 821–823
29. Hassan-Montero Y, Herrero-Solana V (2006) Improving tag-clouds as visual information retrieval interfaces. In: International conference on multidisciplinary information sciences and technologies, pp 25–28, Citeseer
30. Hernández AR, Lorenzo MMG, Simón-Cuevas A, Arco L, Serrano-Guerrero J (2019) A semantic approach for topic-based polarity detection: a case study in the spanish language. Procedia Computer Science 162:849–856
31. Heu J-U, Qasim I, Lee D-H (2015) Fodosu: multi-document summarization exploiting semantic analysis based on social folksonomy. Information Processing & Management 51(1):212–225
32. Houston P (2013) Instant jsoup how-to. Packt Publishing Ltd
33. Hu X, Wu B (2006) Automatic keyword extraction using linguistic features. In: Data Mining Workshops, 2006. ICDM Workshops 2006. Sixth IEEE International Conference on, pp 19–23, IEEE
34. Hulth A (2003) Improved automatic keyword extraction given more linguistic knowledge. In: Proceedings of the 2003 conference on Empirical methods in natural language processing, pp 216–223, Association for Computational Linguistics
35. Ike CS, Muhammad N (2019) Separable property-based super-resolution of lousy image data. Pattern Anal Applic 23(3):1407–1420
36. Kaptein R (2012) Using wordclouds to navigate and summarize twitter search results. In: EuroHCIR, pp 67–70
37. Kasutani E, Yamada A (2001) The mpeg-7 color layout descriptor: a compact image feature description for high-speed image/video segment retrieval. In: Image Processing, 2001. Proceedings. 2001 International Conference on, 1, pp 674–677, IEEE
38. Khalid S, Muhammad N, Sharif M (2018) Automatic measurement of the traffic sign with digital segmentation and recognition. IET Intell Transp Syst 13(2):269–279
39. Khan H, Sharif M, Bibi N, Muhammad N (2019) A novel algorithm for the detection of cerebral aneurysm using sub-band morphological operation. The European Physical Journal Plus 134(1):34
40. Khan H, Sharif M, Bibi N, Usman M, Haider SA, Zainab S, Shah JH, Bashir Y, Muhammad N (2020) Localization of radiance transformation for image dehazing in wavelet domain. Neurocomputing 381:141–151
41. Landauer TK, Foltz PW, Laham D (1998) An introduction to latent semantic analysis. Discourse processes 25(2-3):259–284
42. Mahajani A, Pandya V, Maria I, Sharma D (2019) A comprehensive survey on extractive and abstractive techniques for text summarization. In: Ambient Communications and Computer Systems, pp 339–351, Springer

43. Mahmood Z, Ali T, Muhammad N, Bibi N, Shahzad I, Azmat S (2017) Ear: Enhanced augmented reality system for sports entertainment applications. KSII Transactions on Internet & Information Systems 11(12):6069–6091
44. Mahmood Z, Haneef O, Muhammad N, Khattak S (2018) Towards a fully automated car parking system. IET Intell Transp Syst 13(2):293–302
45. Mathes A (2010) Folksonomies: Cooperative classification and communication through shared metadata, 2004. http://www.adammathes.com/academic/computer-mediated-communication/folksonomies.html
46. Matsuo Y, Ishizuka M (2004) Keyword extraction from a single document using word co-occurrence statistical information. International Journal on Artificial Intelligence Tools 13(01):157–169
47. Mika P (2007) Ontologies are us: A unified model of social networks and semantics. Web semantics: science, services and agents on the World Wide Web 5(1):5–15
48. Miller GA (1995) Wordnet: a lexical database for english. Commun ACM 38(11):39–41
49. Moscato V, Picariello A, Rinaldi AM (2010) A recommendation strategy based on user behavior in digital ecosystems. In: Proceedings of the International Conference on Management of Emergent Digital EcoSystems, pp 25–32, ACM
50. Muhammad N, Bibi N, Jahangir A, Mahmood Z (2018) Image denoising with norm weighted fusion estimators. Pattern Anal Applic 21(4):1013–1022
51. Navigli R, Ponzetto SP (2010) Babelnet: Building a very large multilingual semantic network. In: Proceedings of the 48th annual meeting of the association for computational linguistics, pp 216–225
52. Naz I, Muhammad N, Yasmin M, Sharif M, Shah JH, Fernandes SL (2019) Robust discrimination of leukocytes protuberant types for early diagnosis of leukemia. Journal of Mechanics in Medicine and Biology 19(06):1950055
53. Nguyen K-L, Shin B-J, Yoo SJ (2016) Hot topic detection and technology trend tracking for patents utilizing term frequency and proportional document frequency and semantic information. In: 2016 international conference on big data and smart computing (BigComp), pp 223–230, IEEE
54. Rinaldi AM (2009) An ontology-driven approach for semantic information retrieval on the web. ACM Transactions on Internet Technology (TOIT) 9(3):10
55. Rinaldi AM (2012) Improving tag clouds with ontologies and semantics. In: Database and Expert Systems Applications (DEXA), 2012 23rd International Workshop on, pp 139–143, IEEE
56. Rinaldi AM (2013) Document summarization using semantic clouds. In: Semantic Computing (ICSC), 2013 IEEE Seventh International Conference on, pp 100–103, IEEE
57. Rinaldi AM (2014) A multimedia ontology model based on linguistic properties and audio-visual features. Inf Sci 277:234–246
58. Rinaldi AM (2015) A complete framework to manage multimedia ontologies in digital ecosystems. Int J Bus Process Integr Manag 7(4):274–288
59. Rinaldi AM (2018) A document visualization strategy based on semantic multimedia big data. In: The 16th International Symposium on Pervasive Systems, Algorithms and Networks (I-SPAN 2019), Springer. (in press)
60. Rinaldi AM (2019) Web summarization and browsing through semantic tag clouds. International Journal of Intelligent Information Technologies (IJIIT) 15(3):1–23
61. Rinaldi AM, Russo C (2018) A matching framework for multimedia data integration using semantics and ontologies. In: 2018 IEEE 12th International Conference on Semantic Computing (ICSC), pp 363–368, IEEE
62. Rinaldi AM, Russo C (2018) User-centered information retrieval using semantic multimedia big data. In: 2018 IEEE International Conference on Big Data (Big Data), pp 2304–2313, IEEE
63. Russo C, Madani K, Rinaldi AM (2019) Knowledge construction through semantic interpretation of visual information. In: International Work-Conference on Artificial Neural Networks, pp 246–257, Springer
64. Sala HV, Leyva PR, Febles JP, Sentí VE (2019) Information retrieval with semantic annotation. EasyChair
65. Salatino AA, Osborne F, Thanapalasingam T, Motta E (2019) The cso classifier: Ontology-driven detection of research topics in scholarly articles. In: International Conference on Theory and Practice of Digital Libraries, pp 296–311, Springer
66. Salton G, Singhal A, Mitra M, Buckley C (1997) Automatic text structuring and summarization. Information Processing & Management 33(2):193–207
67. Selvalakshmi B, Subramaniam M (2019) Intelligent ontology based semantic information retrieval using feature selection and classification. Clust Comput 22(5):12871–12881
68. Sinclair J, Cardew-Hall M (2008) The folksonomy tag cloud: when is it useful? J Inf Sci 34(1):15–29
69. VanderWal T (2007) Folksonomy

70. Wei Y (2012) An iterative approach to keywords extraction. In: International Conference in Swarm Intelligence, pp 93–99, Springer
71. Won CS, Park DK, Park S-J (2002) Efficient use of mpeg-7 edge histogram descriptor. ETRI journal 24(1):23–30
72. Xu S, Yang S, Lau FC-M (2010) Keyword extraction and headline generation using novel word features. In: AAAI, pp 1461–1466
73. Yager RR (1988) On ordered weighted averaging aggregation operators in multicriteria decisionmaking. IEEE Transactions on systems, Man, and Cybernetics 18(1):183–190
74. Young T, Hazarika D, Poria S, Cambria E (2018) Recent trends in deep learning based natural language processing. ieee Computational intelligenCe magazine 13(3):55–75
75. Zhang T, Ge SS (2019) An improved tf-idf algorithm based on class discriminative strength for text categorization on desensitized data. In: Proceedings of the 2019 3rd International Conference on Innovation in Artificial Intelligence, pp 39–44
76. Zhu J, Wang C, He X, Bu J, Chen C, Shang S, Qu M, Lu G (2009) Tag-oriented document summarization. In: Proceedings of the 18th international conference on World wide web, pp 1195–1196, ACM