1

Automatic thematic categorization of multimedia documents using ontological information and fuzzy algebra*

Manolis Wallace, Phivos Mylonas, Giorgos Akrivas, Yannis Avrithis, and Stefanos Kollias

Image, Video and Multimedia Laboratory
School of Electrical and Computer Engineering
National Technical University of Athens, Greece
wallace@image.ntua.gr

Summary. The semantic gap is the main problem of content based multimedia retrieval. This refers to the extraction of the semantic content of multimedia documents, the understanding of user information needs and requests, as well as to the matching between the two. In this chapter we focus on the analysis of multimedia documents for the extraction of their semantic content. Our approach is based on fuzzy algebra, as well as fuzzy ontological information. We start by outlining the methodologies that may lead to the creation of a semantic index; these methodologies are integrated in a video annotating environment. Based on the semantic index, we then explain how multimedia content may be analyzed for the extraction of semantic information in the form of thematic categorization. The latter relies on stored knowledge and a fuzzy hierarchical clustering algorithm that uses a similarity measure that is based on the notion of context.

Key words: semantics, semantic multimedia document analysis, fuzzy ontological relations, thematic categorization

1.1 Introduction

In the framework of digital archiving of multimedia content different scientific fields such as database management systems, processing and analysis of multimedia data, as well as artificial and computational intelligence methods, have observed a close cooperation with each other during the last few years. The attempt has been to develop intelligent and efficient human computer interaction systems, enabling the user to access vast amounts of heterogeneous

* This work has been partially funded by the EU IST-1999-20502 FAETHON project and by the "HRAKLEITOS" project. The Project "HRAKLEITOS" is co-funded by the European Social Fund (75%) and National Resources (25%).

information, stored in different sites and archives. The focus of technological attempts has been on the analysis of digital video, due to its large amounts of spatiotemporal interrelations, which turns it into the most demanding and complex data structure. Video understanding and semantic information extraction have been identified as important steps towards more efficient manipulation of visual media. Attaching metadata to the original content is the means for achieving this goal.

Unquestionably, the problem of finding the information that user desires is still an open research problem, although maintaining large archives of digital media is increasingly easy and cost-effective, since cheap storage capacity and network bandwidth are available. The problems of *information overload* (the user is flooded with documents which are irrelevant to the query) and *mismatch* (a document which the user desires does not match the query), which are commonly encountered in textual retrieval systems [19][7] are more severe for multimedia retrieval, as the multimedia content is not in the same form as the query [11] [20]. Thus, a more efficient approach to content-based retrieval is necessary when referring to multimedia documents.

Two main approaches have been utilized for the problem of multimedia information retrieval. The first one, *query by example*, replaces the textual query with an exemplar multimedia document. Visual features are extracted from the example and matched to similar features extracted from every document [10]. Although this approach avoids the problem of matching multimedia to textual terms, it has other difficulties: it is often difficult to find an example of the document one desires, and more importantly, it is not easy to isolate from the example the feature(s) that the user considers relevant to his or her desire. Additionally, in [9] and [8], hybrid methods extending the query by example strategy are developed. The second one, *semantic indexing* [13], retains the textual form of the query and attempts to extract, from the multimedia document, meaningful entities, such as objects and events, which can be mapped to textual terms and then matched to the textual query. This approach makes it possible to adopt successful expansion and adaptation techniques from the textual retrieval paradigm in multimedia retrieval.

Still, many extensions to research completed in textual information retrieval are needed. A semantic approach to content-based multimedia information retrieval will have to tackle at least the following issues:

- analyze metadata and raw multimedia data for the detection of higher level entities, i.e. for the determination of documents' semantics.
- analyze user requests, as to determine their semantic content.
- efficiently match the semantics of user requests to the semantics of documents.

In this chapter we focus on the first of these issues and analyze metadata towards detection of high level semantics. Specifically, starting from a multimedia document, a set of algorithms performing automated spatiotemporal segmentation is implemented. These objects and event detection techniques are integrated in a video annotator application. The user is free to manually select detected objects and events and input free text as annotation, constructing the required semantic index. The application then produces an MPEG-7 annotation that fully describes the hierarchical structure of the document and its semantics at each level of description. Consequently, utilizing semantic ontological information together with a novel definition of context that relies on fuzzy algebra, we estimate the semantic content of the document. The proposed methodology considers and tackles uncertainty in the generation of the semantic index, as well as fuzziness of real life semantics via ontological relations.

The structure of the chapter is as follows. In section 1.2 we provide a description of the annotator application and the semantic index construction. In section 1.3 we present the knowledge representation utilized by our approach, introducing a novel fuzzy quasi-taxonomic relation to model real life information. In section 1.4 we then provide the methodology for detecting thematic categories within multimedia documents. Towards that scope, we utilize the notion of semantic context and propose a fuzzy hierarchical clustering algorithm for the extraction of topics. Finally, section 1.5 provides experimental results and conclusions are drawn in section 1.6.

1.2 Video Analysis, Annotation and Indexing

1.2.1 Multimedia Indexing

In the framework of the MPEG-7 standard [27], features of multimedia content are described through a rich set of data structures, termed Descriptors and Description Schemes. The former define the syntax and the semantics of the representation of features, while the latter specify the structure and semantics of the relationships between Descriptors or other Description Schemes. Low-level object features, such as color, shape, texture, and motion can be described via visual descriptors. Spatiotemporal relations can be described via the segment decomposition and relation tools. High-level features, such as objects, events and abstract concepts, are described via semantic description schemes such as the semantic entity. It is expected that, while most descriptors corresponding to low-level features are extracted automatically, human intervention will be required to produce the high-level descriptors.

Of course, there is no reason to assume that a document's description (either expressed in terms or in semantic entities) contains entities with equal importance. A more flexible and accurate approach is to weight the *importance* of the description terms, i.e. to attach a degree of importance to them. In text documents, a purely semantic degree of importance needs natural language understanding, which is not always feasible; simpler techniques to retrieve the weight of index terms include statistical analysis of terms [3]. Such techniques can also be applied on the textual annotation of multimedia documents. Moreover, in the case of annotation, weights can also be inferred from the underlying structure, e.g. terms found in the title could be weighted more than those found in the free text description. On the other hand, a multimedia document provides some cues that are useful in extracting the objects of interest. Such cues are motion (a person that moves is more important to a shot, than a table, for example), depth (an object in the foreground is in general more important than an object in the background) and context, if available (for example it is important to know if a shot is indoors or outdoors, whether it is in the sea or in a field).

A different source of weights is *uncertainty* [15]. Uncertainty in general plays little role in text indexing, and is measured with techniques such as stemming. Another source of uncertainty in text retrieval is context; for example the same word can have very different meanings depending on context. This is mainly due to the fact that text is not an encapsulated representation of an intended meaning, but rather is a cue that allows the anticipated audience to construct an appropriate meaning heavily based on context. On the other hand, uncertainty in computer vision problems is in general greater, as the descriptors and the models used to describe visual features are imperfect [22]. While importance and uncertainty are very different kinds of information [12], they are correlated, as an object considered to be important for the scene (such as a person in the foreground) is arguably recognized with greater certainty than one that is considered unimportant. Thus, we only use one number for weighting entities.

Still, the problem of semantic analysis of multimedia documents starting from raw multimedia information cannot be tackled in a fully automated manner, when even the problems of semantic segmentation and simple object detection remain open. Thus, we follow a manual approach: we utilize scene detection, object localization and tracking and feature detection techniques to prepare a syntactic description of multimedia content, which is then used by a human to provide textual annotation, at each level of description, using an annotation application and free text analysis.

Different video analysis techniques, such as temporal segmentation, object localization and tracking are integrated in a video annotator application [4]. A screenshot of its user interface is shown in Figure 1.1. This application is able to partition video into shots, cluster similar shots into higher level temporal segments, extract key frames, detect moving objects in a particular shot and let the user annotate the entire video program or each temporal segment, shot, key frame or object by associating it with keywords or free text. A hierarchical spatiotemporal video segmentation is thus achieved, and a textual MPEG-7 annotation is produced as output. This annotation fully describes a multimedia document at a syntactic level. The annotator is able to provide annotation at all levels, from the highest level of the video program or multimedia document to the lowest level of moving objects. This is achieved through the hierarchical structure of video segments, also depicted in Figure 1.1.

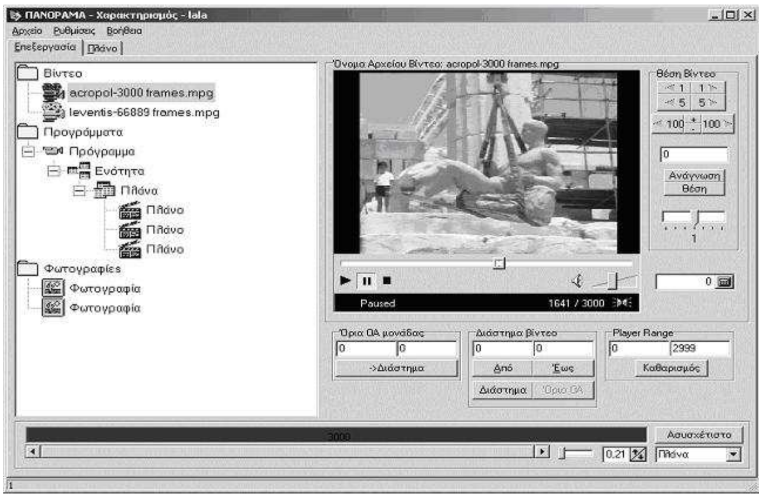

Fig. 1.1. User interface of the annotator application

The main features supported by the annotator application are summarized below:

1. automatic shot detection, shot clustering and key frame extraction
2. object localization and tracking
3. hierarchical representation of multimedia segments (multimedia document $\rightarrow$ video sequence $\rightarrow$ theme $\rightarrow$ shot group $\rightarrow$ shot/key frame $\rightarrow$ moving object)
4. MPEG-7 compliant syntactic description and annotation

The annotator is free to choose the depth in which he decides to annotate. For example, he might annotate only the entire multimedia document or just part or all of the shots. Or he might not take the time to execute the automatic shot detection and limit himself to the annotation only. The overall structure and format of the annotation output is depicted in the MPEG-7 segment presented in Figure 1.2. This annotation consists of characteristic segments, emphasizing on the structure of the corresponding video content. The code provides the description of multimedia audiovisual content. The title of the content is provided, as well as significant information on the particular audiovisual segment: information about the media, its instance locator and uri, information about the creation of the fragment in the form of a title, a free text annotation and a set of keywords, as well as the duration of the audiovisual fragment.

1.2.2 Semantic Indexing

Manually generated MPEG-7 textual metadata need to be treated in a meaningful way, in order to handle user queries during information retrieval. This

```xml
<?xml version="1.0" encoding="ISO-8859-1"?>
<Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="mpeg7 mpeg\Mpeg7-2001.xsd">
    <DescriptionMetadata>
        <CreationTime>1975-10-18</CreationTime>
    </DescriptionMetadata>
    <Description xsi:type="ContentEntityType">
        <MultimediaContent xsi:type="AudioVisualType">
            <AudioVisual>
                <CreationInformation>
                    <Creation>
                        <Title>Konstantinos Karamanlis</Title>
                    </Creation>
                </CreationInformation>
                <TemporalDecomposition>
                    <AudioVisualSegment>
                        <MediaInformation>
                            <MediaProfile>
                                <MediaInstance>
                                    <InstanceIdentifier>1318</InstanceIdentifier>
                                    <MediaLocator>
                                        <MediaUri>Karamanlis.mpg</MediaUri>
                                    </MediaLocator>
                                </MediaInstance>
                            </MediaProfile>
                        </MediaInformation>
                        <CreationInformation>
                            <Creation>
                                <Title>Konstantinos Karamanlis talks to the people</Title>
                                <Abstract>
                                    <FreeTextAnnotation>Konstantinos Karamanlis addresses the crowd from a balcony. the crowd is
holding flags and shouts for his party. Shots of the crowd applausing and shouting.</FreeTextAnnotation>
                                    <KeywordAnnotation>
                                        <Keyword>Crowd</Keyword>
                                        <Keyword>Party</Keyword>
                                    </KeywordAnnotation>
                                </Abstract>
                            </Creation>
                        </CreationInformation>
                        <MediaTime>
                            <MediaTimePoint>T05:29:17</MediaTimePoint>
                            <MediaDuration>P0DT00H03M18S</MediaDuration>
                        </MediaTime>
                    </AudioVisualSegment>
                </TemporalDecomposition>
            </AudioVisual>
        </MultimediaContent>
    </Description>
</Mpeg7>
```


Fig. 1.2. MPEG-7 annotation of a segment of a multimedia document, produced by the annotator application

is achieved through a semantic index, which is automatically generated via recognizing objects and events in multimedia documents, exploiting the textual annotation and mapping it to higher-level semantic entities. Document semantics are subsequently extracted via analysis of the semantic index to be exploited in the retrieval process.

The index contains sets of document locators (links) for each semantic entity (thematic category, object, event, concept, agent, semantic place or semantic time), indicating in which document every entity is encountered. It has the form of a matrix, mapping documents to semantic entities or vice versa. Document locators associated to index entries may link to complete a/v documents, objects, still images or other video decomposition units that may be contained in the a/v archives.

For the automatic generation of the semantic index, semantic interpretation of the free text provided by experts as metadata using the annotator tool is performed, in order to match the textual terms to semantic entities. During this process, a query is issued for all documents that have not been indexed, or whose description has been updated. The textual annotation contained in the MPEG-7 compliant description of each such document is analyzed and semantic entities are identified through matching with their definitions in the knowledge model. Links between detected semantic entities and the document in question are then added to the index; weights are also added depending on the location of each entity in the description and the degree of entity's matching.

The definitions of semantic entities in the knowledge model may contain sequences of labels, each one providing a different textual form of the semantic entity, possibly in more than one language. Matching those to the terms in the textual annotation can provide the semantic representation of the annotation. Of course, in most cases this is far from trivial: the mapping between terms and semantic entities is a many-to-many relation, which means that multiple possible semantic interpretations exist for the same text.

As a simple example, consider the case of the term ‘element‘. At least two distinct semantic entities correspond to it: ‘element1‘, which is related to chemistry, and ‘element2‘, which is related to XML. Supposing that the annotation contains the term ‘element‘, we need to determine which semantic entity the term should be mapped to. In the same example, if the remaining terms of the annotation are related to chemistry, then it is quite safe to suppose that the user is referring to semantic entity ‘element1‘ rather than to semantic entity ‘element2‘. This implies that the context of the text can be used to facilitate the process of semantic entity determination in the case of ambiguities. However, the context can only be estimated using a representation of a fuzzy set of entities, and thus cannot be performed before the interpretation is completed. Consequently, interpretation needs to take place simultaneously with context estimation.

1.3 Knowledge Model for Semantic Document Analysis

Once the semantic index has been constructed, semantic document analysis and clustering is carried out in order to detect thematic categories in each multimedia document. This section provides the necessary background for the further analysis steps. In particular, we introduce a novel fuzzy quasi-taxonomic relation, constructed through a combination of suitable relations, in order to best handle knowledge and context representation.

The approach followed is based on a formal methodology that is founded on fuzzy relational algebra [14]. The mathematical notation used for this purpose is summarized in the following.

1.3.1 Ontological relations

It is common knowledge that retrieval systems based on terms suffer from the problematic mapping of terms to concepts [18]. Specifically, as more than one terms may be associated to the same concept, and more than one concepts may be associated to the same term, the processing of query and index information is not trivial. In order to overcome such problems, we have chosen to work directly with concepts, rather than terms. We refer to these concepts as *semantic entities* [18]. In the sequel, we will denote by $S = \{s_1, s_2, \ldots, s_n\}$, the set of semantic entities that are known. The definitions of these entities, together with their textual descriptions, i.e. their corresponding terms, reside in a *semantic encyclopedia*. An encyclopedia additionally contains semantic relations amongst the semantic entities. Recently, much attention has been drawn to the design, representation, population and storage of such relations, using ontologies [16]. An ontology can be viewed as a framework for knowledge representation in which the context determines the intended meaning of each word. A word used in different context may have different meanings. In general, ontologies may be described as follows:

$$O = \{S, \{R_i\}\}, i = 1 \ldots n \tag{1.1}$$

$$R_i : S \times S \rightarrow \{0, 1\}, i = 1 \ldots n \tag{1.2}$$

where O is an ontology and R_i the i-th semantic relation amongst the semantic entities. The formal definition of ontologies also supports an inference layer, but this is outside the scope of this chapter.

Although any type of relation may be contained in an ontology, the two main categories are taxonomic (i.e. ordering) and compatibility (i.e. symmetric) relations. Compatibility relations have traditionally been exploited by information retrieval systems for tasks such as query expansion. They are ideal for the description of similarities of various natures, but fail to assist in the determination of the context of a query or a document; the use of ordering relations is necessary for such tasks [2]. Thus, a main challenge of intelligent multimedia retrieval is the meaningful exploitation of information contained in taxonomic relations of an ontology.

It is well understood that relations among real life entities are always a matter of degree, and are, therefore, best modelled using fuzzy relations. A rather self-explicable example is presented in Figure 1.3. Ontological taxonomies, on the other hand, are crisp in principle. Thus, they fail to fully describe real life concepts, and are limited to α-cuts of the desired relations. In order to explain the difference, consider A as a fuzzy subset of X, where X is a space of points or what is called a “Universe of Discourse”. Fuzzy Set A in X is characterized by a membership function $\mu_A(x)$ which associates a real number in the interval $[0, 1]$ with each point x in X. Then its α-cut A_a is a non-fuzzy subset of X defined by $A_a = \{x | \mu_A(x) \geqq \alpha\}$. Figure 1.4 shows an α-cut on fuzzy set A.
This is a very important drawback, that makes such relations insufficient for the services that an intelligent multimedia retrieval system aims to offer.

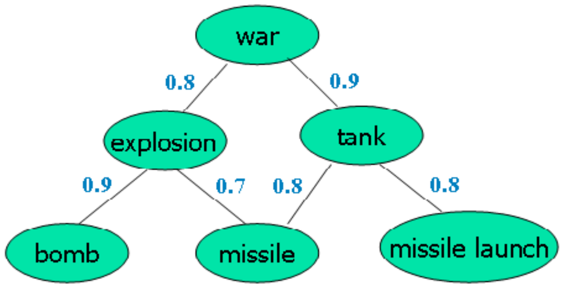

Fig. 1.3. Each relation is dominated by a degree d, which could be crisp (0 or 1) or fuzzy (i.e. 0.8)

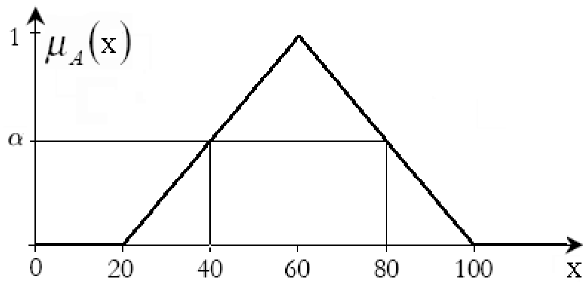

Fig. 1.4. α-cut on a fuzzy set A

1.3.2 The Fuzzy Quasi – Taxonomic Relation

In order to overcome such problems, fuzzy semantic relations have been proposed for the modelling of real life information [1]. In this section, we present a few commonly encountered semantic relations that can be modelled as fuzzy ordering relations, and propose their combination for the generation of a meaningful, fuzzy, quasi-taxonomic relation. Based on this relation, we will explain how the problem of automatic thematic categorization may be tackled in section 4.

The *specialization* relation Sp is a fuzzy partial ordering relation on the set of semantic entities. $Sp(a, b) > 0$ means that the meaning of a "includes" the meaning of b; the most common form of specialization is sub-classing, i.e. a is a generalization of b. For example a could be a vehicle and b could be a car. The role of the specialization relation in knowledge-based retrieval is as follows: if a document refers to the meaning of entity b, then it is also related to a, since b is a special case of a. Still, there is no evidence that the opposite also holds; it is obvious that the specialization relation contains important information that cannot be modelled in a symmetric relation.

The *part* relation P is also a fuzzy partial ordering on the set of semantic entities. $P(a, b) > 0$ means that b is a part of a. For example a could be a human body and b could be a hand. The role of P in content-based retrieval is the opposite of that of Sp; if the user query contains b, then a document containing a will probably be of interest, because a contains a part b.

The *context* relation Ct is also a fuzzy partial ordering on the set of semantic entities. $Ct(a, b) > 0$ means that b provides the context for a or, in other words, that b is the thematic category that a belongs to. Other relations considered in the following have similar interpretations. Their names and corresponding notations are given in Table 1.1.

Table 1.1. The fuzzy semantic relations

Sp	Specialization
Ct	Context
Ins	Instrument
P	Part
Pat	Patient
Loc	Location
Pr	Property

In this work, fuzziness of the aforementioned relations has the following meaning: High values of $Sp(a, b)$, imply that the meaning of b approaches the meaning of a, in the sense that when a document is related to b, then it is most probably related to a as well. On the other hand, as $Sp(a, b)$ decreases, the meaning of b becomes "narrower" than the meaning of a, in the sense that a document's relation to b will not imply a relation to a as well with a high probability, or to a high degree. Summarizing, the value of $Sp(a, b)$ indicates the degree to which the stored knowledge shows that an occurrence of b in a document implies relation to a. Likewise, the degrees of the other relations can also be interpreted as conditional probabilities or degrees of implied relevance.

The above imply that, for example, $a \neq b \rightarrow Sp(a, b) < 1$ since, if $a \neq b$, then we cannot be sure that both a and b are related to a given document, without first examining the document's context; at this point it is important to remind that a and b are not terms but concepts, which means that $a \neq b$ indicates / ensures a difference in a conceptual level.

A last point to consider is the transitivity of the relations presented above. It is obvious that if b is a specialization of a and c is a specialization of b, then c is a specialization of a. This implies that the specialization relation is transitive. A similar argument can be made for the other relations, as well. Let us now consider a more practical example. Let a be the concept of "car", b the concept of "wheel" and c the concept of "rubber". The inclusion $a < b < c$ is rather obvious. Still, it is not equally obvious that a user requesting documents related to rubber will be satisfied when faced with documents that are related to cars. By this example we wish to demonstrate that the form of transitivity used cannot be $max - min$ transitivity, but one relying on a subidempotent norm. Therefore, we demand that the presented relations are sup-t transitive, where t is an Archimedean norm. This means that $Sp(a,c) \geq \max_{s \in S} t(Sp(a,s), Sp(s,c))$, $t(a,a) < a$ and, therefore, $t(a,b) < \min(a,b), \forall a \in (0,1)$.

More formally, the knowledge model presented above may be summarized in the following:

$$O_{\mathcal{F}} = \{S, \{r_i\}\}, i = 1 \ldots n \tag{1.3}$$

$$r_i = \mathcal{F}(R_i) : S \times S \to [0,1], i = 1 \ldots n \tag{1.4}$$

The existence of many relations has lead to the need for utilization of more relations for the generation of an adequate taxonomic relation T. Based on the relations r_i we construct the following semantic relation:

$$T = Tr^t(\bigcup_i r_i^{p_i}), p_i \in \{-1, 1\}, i \in 1 \ldots n \tag{1.5}$$

where $Tr^t(A)$ is the sup-t transitive closure of relation A; the transitivity of relation T was not implied by the definition, as the union of transitive relations is not necessarily transitive. In this work we use a taxonomic relation that has been generated with the use of the following semantic relations:

- Specialization *Sp*.
- Context *Ct*, inverted.
- Part *P*.
- Instrument *Ins*. $Ins(a,b) > 0$ indicates that b is an instrument of a. For example, a may be “music” and b may be “drums”.
- Location *Loc*. $Loc(a,b) > 0$ indicates that b is the location of a. For example, a may be “concert” and b may be “stage”.
- Patient *Pat*. $Pat(a,b) > 0$ indicates that b is a patient of a. For example, a may be “course” and b may be “student”.
- Property *Pr*. $Pr(a,b) > 0$ indicates that b is a property of a. For example, a may be “Jordan” and b may be “star”.

Thus, the utilized relation is:

$$T = Tr^t(Sp \cup Ct^{-1} \cup Ins \cup P \cup Pat \cup Loc \cup Pr) \tag{1.6}$$

An example of the T relation taken from the airplane domain is presented in Figure 1.5. Based on the semantics of relations r_i, it is easy to see that T is ideal for the determination of the thematic categories that an entity may be related to, as thematic categories are also semantic entities:

$$C \subseteq S \tag{1.7}$$

where $C = \{c_i\}, i \in 1 \ldots k$ is the set of thematic categories (for example *ball* and *stadium* may be semantic entities, while *football* and *sports* are both semantic entities and thematic categories).

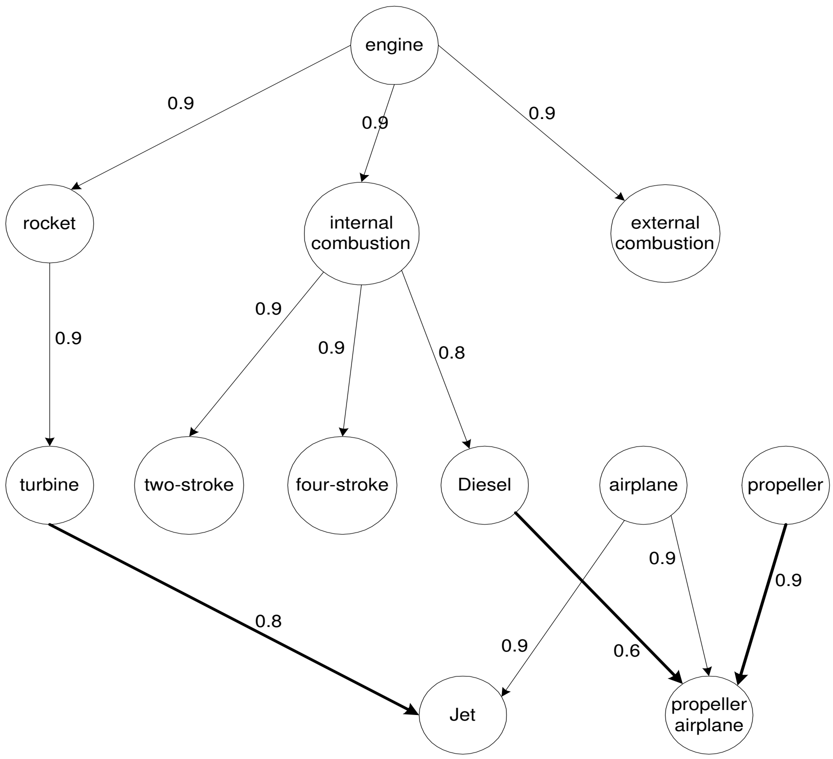

Fig. 1.5. Example of T relation construction

All the relations used for the generation of T are partial ordering relations. Still, there is no evidence that their union is also antisymmetric. Quite the contrary, T may vary from being a partial ordering to being an equivalence relation. This is an important observation, as true semantic relations also fit in this range (total symmetricity as well as total antisymmetricity often have to be abandoned when modelling real life). Still, the semantics of the used relations, as well as our experiments, indicate that T is “almost” antisymmetric. Therefore, we categorize to it as quasi-ordering or quasi-taxonomic.

1.4 Detection of Thematic Categories

In this section we focus on the extraction of semantic content of multimedia documents, in the form of thematic categorization. Specifically, we present initially the main problem formulation, whose aim is to detect which semantic entities and thematic categories are related to a particular document. In the following, we explain how the notion of context can be defined using the aforementioned fuzzy quasi-taxonomic relation. Continuing, we explain how this context can be utilized to detect the thematic categories to which a document is related, while at the same time overcoming the problem of uncertainty or noise in the semantic index, based on a fuzzy hierarchical clustering approach.

1.4.1 Problem formulation

Let us first present the problem that this work attempts to address, in a more formal manner. The main objective is to analyze the semantic index, with the aim of extracting a document's semantics. In other words, we aim to detect which semantic entities and thematic categories are indeed related to a document, and to which extent. More formally, we accept as input the semantic indexing of available documents, i.e. the semantic index I. This is in fact a fuzzy relation between the sets of documents D and semantic entities S:

$$I : D \times S \to [0, 1] \tag{1.8}$$

Each document d is represented as a normal fuzzy set $I(d)$ on the set of semantic entities, i.e.:

$$\forall d \in D \;\; \exists s \in S \text{ such that } I(s, d) = 1$$

Based on this set, and the knowledge contained in the available semantic relations, we aim to detect the degree to which a given document d is related to a semantic entity $s \in S$. This entity may be (and usually is) new with respect to set $I(d)$, i.e. it may not be already known to be associated with document d simply based on the document indexing process. We will refer to this degree as $R_T(s, d)$. In other words, we attempt to calculate a relation

$$R_T : S \times D \to [0, 1] \tag{1.9}$$

where D is the set of available documents, as already explained. In designing an algorithm that is able to calculate this relation in a meaningful manner, a series of issues need to be tackled. Examples of such issues are depicted in Figure 1.6.

Among others we observe that:

- there are no topics that relate all the entities
- entity 1 is related to topics A, B and C, but only the latter two are related to the whole document, due to their numerous (> 1) relations to distinct topics within the whole document. Also, topic A is not necessarily related to the whole document.

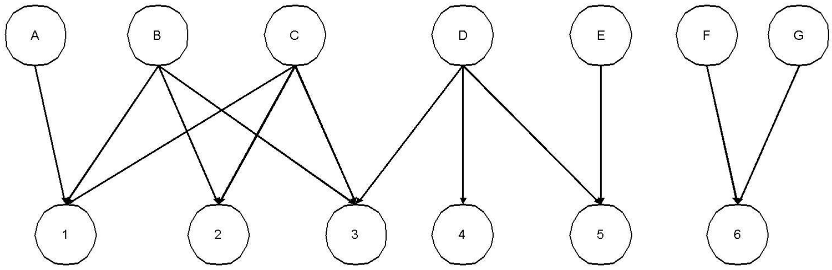

Fig. 1.6. Entities that index a document (1,2,..,6), related topics detected (A,B,...,G) and relations among them.

- entity 3 is related to two distinct topics of interest. Topics B and C are considered a team, since they relate to the exact same entities
- topics F, G are related to only one of the document's entities; this could be coincidental.

Consequently, the above example illustrates all the issues to be tackled in designing an efficient algorithm, which can be summarized into the following:

1. A semantic entity may be related to multiple, unrelated topics. Example: a ball may be related to baseball, basketball, kids ball, etc. Consequently, the common meaning of the remaining entities that index the given document has to be considered.
2. A document may be related to multiple, unrelated topics. Example: fans in a stadium may imply football match, concert, protest, etc. If this is the case, most entities will be related to just one of these topics. Therefore, clustering of the remaining entities, based on their common meaning, needs to be applied.
3. The semantic index may contain incorrectly recognized entities. Example: entities from the use of terms in metaphorical sense. Those entities will not be found similar to other entities, so the cardinality of the clusters can be used.

In the following, keeping these issues in mind, we provide the principles of the applied approach. According to issue (1), a semantic entity may correspond to multiple, unrelated topics. Therefore, it is necessary for the algorithm to be able to determine which of these topics are indeed related to a given document. In order for this task to be performed in a meaningful manner, the common meaning of the remaining entities that index the given document needs to be considered as well.

On the other hand, when a document is related to more than one, unrelated topics, as issue (2) points out, we should not expect all the entities that index it to be related to each one of the topics in question. Quite the contrary, we should expect most entities to be related to just one of these them. Therefore, a clustering of semantic entities, based on their common meaning, needs to be applied.

In this process, entities that are misleading (e.g. entities that resulted from incorrect detection of entities in the document) will probably not be found similar with other entities that index the given document. Therefore, the cardinality of the clusters may be used to tackle issue (3).

The proposed approach may be decomposed into the following steps:

1. Create a single taxonomic semantic relation that is suitable for use by the thematic categorization module.
2. Determine the count of distinct topics that a document is related to, by performing a partitioning of semantic entities, using their common meaning as clustering criterion.
3. Fuzzify the partitioning, in order to allow for overlapping of clusters and fuzzy membership degrees.
4. Identify the topic that is related to each cluster.
5. Aggregate the topics for distinct clusters in order to acquire an overall result for the document.

Each of the above steps uses the taxonomy relation, in addition to the index. In the following, after discussing the notion of "common meaning", we elaborate on each of these steps.

1.4.2 The notion of context

We have shown that in the process of content analysis we have to use the common meaning of semantic entities. We will refer to this as their *context* [2]; in general, the term context refers to whatever is common among a set of elements. Relation T will be used for the detection the context of a set of semantic entities, as explained in the remaining of this subsection.

A document d is represented only by its mapping to semantic entities, via the semantic index. Therefore, the context of a document is again defined via the semantic entities that are related to it. The fact that relation T described in subsection 1.3.2 is (almost) an ordering relation allows us to use it in order to define, extract and use the context of a document, or a set of semantic entities in general.

Relying on the semantics of relation T, we define the context $K(s)$ of a single semantic entity $s \in S$ as the set of its antecedents in relation T. More formally, $K(s) = T(s)$. Assuming that a set of entities $A \subseteq S$ is crisp, i.e. all considered entities belong to the set with degree one, the context of the group, which is again a set of semantic entities, can be defined simply as the set of their common antecedents:

$$K(A) = \bigcap_i K(s_i), s_i \in A \tag{1.10}$$

Obviously, as more entities are considered, the context becomes narrower, i.e. it contains less entities and to smaller degrees (Figure 1.7):

$$A \supset B \rightarrow K(A) \subseteq K(B) \tag{1.11}$$

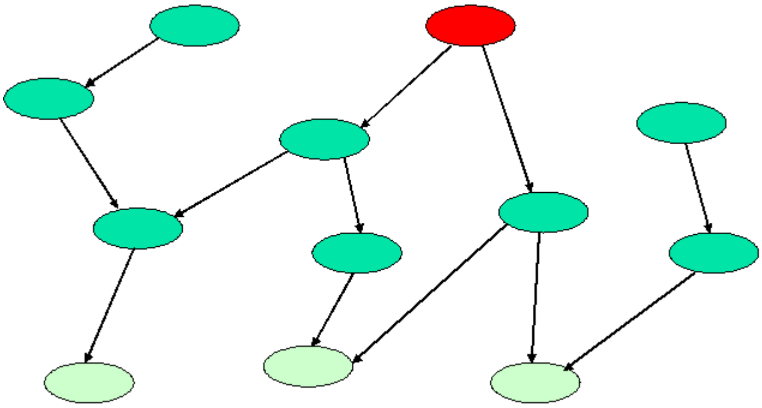

Fig. 1.7. As more entities are considered, the context it contains less entities and to smaller degrees: Considering only the first two leaves from the left, the context contains two entities, whereas considering all the leaves narrows the context to just one common descendant

When the definition of context is extended to the case of fuzzy sets of semantic entities, i.e., A is fuzzy, this property must still hold. Moreover, we demand that the following are satisfied as well, basically because of the nature of fuzzy sets:

- $A(s) = 0 \implies K(A) = K(A - \{s\})$, i.e. no narrowing of context.
- $A(s) = 1 \implies K(A) \subseteq K(s)$, i.e. full narrowing of context.
- $K(A)$ decreases monotonically with respect to $A(s)$.

Taking these into consideration, we demand that, when A is a normal fuzzy set, the "considered" context $\mathcal{K}(s)$ of s, i.e. the entity's context when taking its degree of participation to the set into account, is low when the degree of participation $A(s)$ is high, or when the context of the crisp entity $K(s)$ is low. Therefore

$$\mathcal{K}(s) \doteq K(s) \cup cp(S \cdot A(s)) \tag{1.12}$$

where cp is an involutive fuzzy complement and $S \cdot A(s)$ is a fuzzy set for which

$$[S \cdot A(s)]_{(x)} = A(s) \forall x \in S \tag{1.13}$$

Then the set's context is easily calculated as

$$K(A) = \bigcap_i \mathcal{K}(s_i), s_i \in A \tag{1.14}$$

Considering the semantics of the T relation and the process of context determination, it is easy to realize that when the entities in a set are highly related to a common meaning, the context will have high degrees of membership for the entities that represent this common meaning. Therefore, we introduce the height of the context $h(K(A))$, which may be used as a measure of the semantic correlation of entities in fuzzy set A. We will refer to this measure as *intensity* of the context. The intensity of the context demonstrates the degree of relevance, as shown in Figure 1.8.

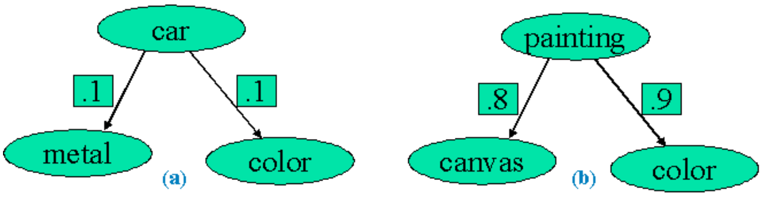

Fig. 1.8. Examples of different heights of context: (a) In the first set of entities, the degree of relevance is rather small and equal, so the height of the context is also small. (b) The second set of entities presents a differentiation in the degree of relevance between the two entities, so the height of the context is greater.

1.4.3 Fuzzy hierarchical clustering and topic extraction

Before detecting the topics that are related to a document d and in order to support the possibility of existence of multiple distinct topics in a single document, the set of semantic entities that are related to it needs to be clustered, according to their common meaning. More specifically, the set to be clustered is the support of the document:

$$^{0+}d = \{s \in S : I(s, d) > 0\} \tag{1.15}$$

Most clustering methods belong to either of two general categories, partitioning and hierarchical [21]. Partitioning methods create a crisp or fuzzy clustering of a given data set, but require the number of clusters as input. Since the number of topics that exist in a document is not known beforehand, partitioning methods are inapplicable for the task at hand [17]; a hierarchical clustering algorithm needs to be applied.

Hierarchical methods are divided into agglomerative and divisive. Of those, the first are more widely studied and applied, as well as more robust. Their general structure, adjusted for the needs of the problem at hand, is as follows:

1. When considering document d, turn each semantic entity $s \in^{0+} d$ into a singleton, i.e. into a cluster c of its own.
2. For each pair of clusters c_1, c_2 calculate a compatibility indicator $CI(c_1, c_2)$. The CI is also referred to as cluster similarity, or distance metric.
3. Merge the pair of clusters that have the best CI. Depending on whether this is a similarity or a distance metric, the best indicator could be selected using the *max* or the *min* operator, respectively.
4. Continue at step 2, until the termination criterion is satisfied. The termination criterion most commonly used is the definition of a threshold for the value of the best compatibility indicator.

The two key points in hierarchical clustering are the identification of the clusters to merge at each step, i.e. the definition of a meaningful metric for CI, and the identification of the optimal terminating step, i.e. the definition of a meaningful termination criterion.

When clustering semantic entities, the ideal distance metric for two clusters c_1, c_2 is one that quantifies their semantic correlation. In the previous subsection we have defined such a metric, the intensity of their common context $h(K(c_1 \cup c_2))$. Therefore, the process of merging of clusters will be based on this measure and should terminate when the entities are clustered into sets that correspond to distinct topics. We may identify such sets by the fact that their common contexts will have low intensity. Therefore, the termination criterion shall be a threshold on the selected compatibility metric.

Hierarchical clustering methods are more flexible than their partitioning counterparts, in that they do not need the number of clusters as an input. This clustering method, being a hierarchical one, will successfully determine the count of distinct clusters that exist in ^{0+}d. Still, it is less robust and inferior to partitioning approaches in the following senses:

- It only creates crisp clusters, i.e. it does not allow for degrees of membership in the output.
- It only creates partitions, i.e. it does not allow for overlapping among the detected clusters.

Both of the above are great disadvantages for the problem at hand, as they are not compatible with the task's semantics: in real life, a semantic entity may be related to a topic to a degree other than 1 or 0, and may also be related to more than one distinct topics. In order to overcome such problems, we describe in the following a method for fuzzification of the partitioning. Thus, the clusters' scalar cardinalities will be corrected, so that they may be used later on for the filtering of misleading entities.

Each cluster is described by the crisp set of semantic entities $c \subseteq^{0+} d$ that belong to it. Using those, we may construct a fuzzy classifier, i.e. a function C_c that measures the degree of correlation of a semantic entity s with cluster c.

$$C_c : S \to [0, 1]$$

Obviously, a semantic entity s should be considered correlated with c, if it is related to the common meaning of the semantic entities in c. Therefore, the quantity

$$C_1(c, s) = h(K(c \cup \{s\})) \tag{1.16}$$

where $h(\cdot)$ symbolizes the height of a fuzzy set, is a meaningful measure of correlation. Of course, not all clusters are equally compact; we may measure cluster compactness using the similarity among the entities it contains, i.e. using the intensity of the cluster's context. Therefore, the aforementioned correlation measure needs to be adjusted, to the characteristics of the cluster in question:

$$C_2(c, s) = \frac{C_1(c, s)}{h(K(c))}$$

It is easy to see that this measure obviously has the following properties:

- $C_2(c, s) = 1$ if the semantics of s imply it should belong to c. For example $C_2(c, s) = 1, \forall s \in c$
- $C_2(c, s) = 0$ if the semantics of s imply it should not belong to c.
- $C_2(c, s) \in (0, 1)$ if s is neither totally related, nor totally unrelated to c.

These are the properties that we wish for the cluster's fuzzy classifier, so we define the correlation of s with c as:

$$C_c(s) \doteq C_2(c, s) = \frac{C_1(c, s)}{h(K(c))} = \frac{h(K(c \cup \{s\}))}{h(K(c))} \tag{1.17}$$

Using such classifiers, we may expand the detected crisp partitions to include more semantic entities. Cluster c is replaced by the fuzzy cluster c^f:

$$c^f = \sum_{s \in {}^{0+}d} s / C_c(s) \tag{1.18}$$

Obviously $c^f \supseteq c$.

The process of fuzzy hierarchical clustering has been based on the crisp set ${}^{0+}d$, thus ignoring fuzziness in the semantic index. In order to incorporate this information when calculating the clusters that describe a document's content, we adjust the degrees of membership for them as follows:

$$c^i(s) = t(c^f(s), I(s, d)), \forall s \in {}^{0+}d \tag{1.19}$$

where t is a t-norm. The semantic nature of this operation demands that t is an Archimedean norm. Each one of the resulting clusters corresponds to one of the distinct topics of the document.

In order to determine the topics that are related to a cluster c^i, two things need to be considered: the scalar cardinality of the cluster $|c^i|$ and its context.

Since context has been defined only for normal fuzzy sets, we need to first normalize the cluster as follows:

$$c^n(s) = \frac{c^i(s)}{h(c^i(s))}, \forall s \in {}^{0+}d \tag{1.20}$$

Obviously, semantic entities that are not contained in the context of c^n cannot be considered as being related to the topic of the cluster. Therefore

$$R_T(c^i) \subseteq R_T^*(c^n) = w(K(c^n)) \tag{1.21}$$

where w is a weak modifier. Modifiers, which are also met in the literature as *linguistic hedges* [14], are used in this work to adjust mathematically computed values so as to match their semantically anticipated counterparts.

In the case where the semantic entities that index document d are all clustered in a unique cluster c^i, then $R_T(d) = R_T^*(c^n)$ is a meaningful approach, where R_T^* corresponds to the output in case of neglecting cluster cardinality. On the other hand, when more than one clusters are detected, then it is imperative that cluster cardinalities are considered as well.

Clusters of extremely low cardinality probably only contain misleading entities, and therefore need to be ignored in the estimation of $R_T(d)$. On the contrary, clusters of high cardinality almost certainly correspond to the distinct topics that d is related to, and need to be considered in the estimation of $R_T(d)$. The notion of "high cardinality" is modelled with the use of a "large" fuzzy number $L(\cdot)$, which forms a function from the set of real positive numbers to the $[0, 1]$ interval, quantifying the notion of "large" or "high". Accordingly, $L(a)$ is the truth value of the proposition "a is high", and, consequently, $L(|b|)$ is the truth value of the proposition "the cardinality of cluster b is high".

The topics that are related to each cluster are computed, after adjusting membership degrees according to scalar cardinalities, as follows:

$$R_T(c^i) = R_T^*(c^n) \cdot L(|c^i|) \tag{1.22}$$

The set of topics that correspond to a document is the set of topics that belong to any of the detected clusters of semantic entities that index the given document.

$$R_T(d) = \bigcup_{c^i \in G} R_T(c^i) \tag{1.23}$$

where $\bigcup$ is a fuzzy co-norm and G is the set of fuzzy clusters that have been detected in d. It is easy to see that $R_T(s,d)$ will be high if a cluster c^i, whose context contains s, is detected in d, and additionally, the cardinality of c^i is high and the degree of membership of s in the context of the cluster is also high (i.e., if the topic is related to the cluster and the cluster does not comprised of misleading entities).

1.5 Examples and Results

A first experiment for the validation of the proposed methodology is presented in the sequel, involving the thematic categorization of five multimedia documents. The documents have been processed and manually annotated using the tools presented in section 2. A limited set of semantic entities have been automatically extracted from the textual annotation to construct the semantic index.

The semantic entities included in the manually constructed, limited knowledge base for this purpose are shown in Table 1.2, with thematic categories shown in boldface. The taxonomy relation available is shown in Table 1.3, using the entity mnemonics of Table 1.2. Zero elements of the relations, as well as elements that are implied by reflexivity are omitted. A portion of the semantic index constructed for the five documents is shown in Table 1.4, where the entities detected in document d_5 are omitted from the table and presented in the text below. Finally, the results of the algorithm for the detection of thematic categories in the documents are shown in Table 1.5.

Table 1.2. Semantic Entity names

S.Entity	*Mnemonic*	*S.Entity*	*Mnemonic*
arts	**art**	army or police uniform	unf
tank	tnk	lawn	lwn
missile	msl	goal	gol
scene	scn	shoot	sht
war	**war**	tier	tir
cinema	**cnm**	river	riv
performer	prf	speak	spk
sitting person	spr	F16	f16
explosion	exp	football player	fpl
launch of missile	lms	goalkeeper	glk
screen	scr	**theater**	**thr**
football	**fbl**	fighter airplane	far
curtain	crn	seat	sit

Document d_1 contains a shot of a theater hall. The play is related to war. We can see that objects and events are detected with a limited degree of certainty. Furthermore, detected entities are not always directly related to the overall topic of the document (for example a “tank” may appear in a shot from a theater, as a part of the play, but this is not a piece of information that can aid in the process of thematic categorization). The algorithm of document analysis ignores “tank” and “speak”.

Document d_2 contains a shot from a cinema hall. The film is again related to war. Although some entities are common between d_1 and d_2 (and they are related to both “theater” and “cinema”), the algorithm correctly detects that

Table 1.3. The taxonomy relation

s_1	s_2	$T(s_1,s_2)$	s_1	s_2	$T(s_1,s_2)$	s_1	s_2	$T(s_1,s_2)$
war	unf	0.90	war	exp	0.60	war	lms	0.70
war	far	0.80	fbl	gol	0.80	fbl	lwn	0.90
war	tnk	0.80	fbl	sit	0.60	cnm	scr	0.90
war	msl	0.80	cnm	sit	0.60	cnm	spr	0.80
thr	scn	0.90	fbl	sht	0.90	fbl	spr	0.60
thr	prf	0.90	fbl	tir	0.80	thr	sit	0.60
thr	spr	0.80	fbl	fpl	0.90	thr	crn	0.70
far	f16	1.00	art	cnm	0.80	art	thr	0.80
fpl	glk	1.00						

Table 1.4. The Semantic Index

s	$d_1(s)$	s	$d_2(s)$	s	$d_3(s)$	s	$d_4(s)$
prf	0,9	spr	0.9	spr	0.8	spr	0.2
spr	0.9	spk	0.8	unf	0.9	unf	0.3
spk	0.6	sit	0.9	lwn	0.6	lwn	0.4
sit	0.7	scr	1.00	gol	0.9	gol	0.3
crn	0.8	tnk	0.4	tir	0.7	tir	0.4
scn	0.9			spk	0.9	spk	0.2
tnk	0.7			glk	0.6	glk	0.3
				sht	0.5	sht	0.4

Table 1.5. The result of semantic document analysis

	$R_T(d_1)$	$R_T(d_2)$	$R_T(d_3)$	$R_T(d_4)$	$R_T(d_5)$
arts	0.84	0.73			0.85
cinema		0.74			0.86
theater	0.89				0.33
football			0.84	0.37	0.77
war					0.77

in this case the overall topic is different. This is accomplished by considering that “screen” alters the context and thus the overall meaning.

Documents d_3 and d_4 are both related to football. Their difference is the certainty with which entities have been detected in them. As can be seen, the algorithm successfully incorporates uncertainty of the input in its result.

As a last example, document d_5 is a sequence of shots from a news broadcast. Due to the diversity of stories presented in it, the semantic entities that are detected and included in the index are quite unrelated to each other. Using the sum notation for fuzzy sets,

d_5= spr/0.9 + unf/0.8 + lwn/0.5 + gol/0.9 + tir/0.7 + spk/0.9 + glk/0.8 + sht/0.5 + prf/0.7 + sit/0.9 + crn/0.7 + scn/0.8 + tnk/0.9 + msl/0.8 + exp/0.9 + riv/1

After the consideration of the fuzziness of the index, the following five fuzzy clusters of entities are created:

c_1= spk/0.9

c_2= riv/1.0

c_3= spr/0.9 + prf/0.7 + sit/0.77 + crn/0.7 + scn/0.8

c_4= spr/0.9 + lwn/0.5 + gol/0.9 + tir/0.7 + glk/0.8 + sht/0.5 + sit/0.9

c_5= unf/0.8 + tnk/0.9 + msl/0.8 + exp/0.9

We can observe that the algorithm successfully identifies the existence of more than one distinct topics in the document. Furthermore, entities such as "seat" and "sitting-person" are assigned to more than one clusters, as they are related to more than one of the contexts that are detected in the document. In the following steps of the algorithm, the first two clusters are ignored, due to their small scalar cardinality.

The methodology described so far has been used in the design and the implementation of the Detection of Thematic Categories (DTC) module, an internal intelligent module of the Faethon multimedia mediator system [6]. In the Faethon system, the role of the DTC module is to parse document annotations and provide thematic categorization for them; this is then used in order to facilitate browsing, searching and personalization tasks. The mediator system integrates five archives, different in architecture, content and annotation language [23]. These are ERT (the Hellenic Public Broadcasting Corporation), FAA (Film Archive Austria), Alinari Archive (Italy), ORF (Austria) and FAG (Film Archive Greece). In the working prototype of the system each archive participates with approximately 200 documents, resulting in a total number of 1005 annotated multimedia documents [24]. WordNet synsets have been used as a source for the definition of the core body of the knowledge base semantic entities, resulting in over 70000 semantic entities.

The list of semantic entities that are characterized as thematic categories appears in the first column of Table 1.6. The second through sixth columns of Table 1.6 present the count of documents from each archive that match each thematic category; as the estimated relevance of documents to thematic categories using the methodology of this chapter is a matter of degree, a threshold of $T_c = 0.6$ is used in order to acquire crisp estimations of thematic categorization. We can see that, although some archives (e.g. FAA) have more documents related to sports while others (e.g. ERT) more related to military issues, otherwise all archives contain documents related to most thematic categories. The last column of the table presents the total count of documents (considering all five archives) that are related to each thematic category. It can be seen that documents map to the thematic categories in a rather uniform way, which makes the thematic categorization a powerful tool for retrieval and personalization tasks [24], [25].

In Table 1.7 we present the distribution of documents to thematic categories, i.e., the count of documents that are not related to any thematic categories, are related to exactly one thematic category, to exactly two thematic categories and so on. The majority of documents are related to multiple

Table 1.6. Thematic categories and archives

	ERT	*FAA*	*Alinari*	*ORF*	*FAG*	*Total*
business	14	6	54	39	6	119
history	78	26	23	24	195	346
olympics	12	87	45	16	14	174
football	5	181	144	68	68	466
sports	19	127	96	40	43	325
basketball	7	96	168	77	79	427
news	111	32	124	164	123	554
military	176	5	80	101	165	527
swimming	24	58	64	24	36	206
tennis	23	114	36	97	12	282
theater	98	2	67	69	45	281
politics	126	15	22	95	133	391
arts	135	13	96	74	91	409
commerce	35	26	23	11	17	112
technology	43	32	5	64	3	147
entertainment	55	45	175	137	74	486
health	63	12	34	42	35	186
education	23	7	76	32	2	140
music	78	33	36	92	66	305
cinema	98	24	165	149	52	488
nature	34	4	74	46	10	168
science	9	26	55	60	16	166
war	139	14	88	121	87	449

thematic categories - typically from 4 to 8 - which validates our fuzzy clustering approach; without it, classification of a document to multiple thematic categories would not have been possible.

Table 1.7. Distribution of documents to thematic categories

Categories	*None*	1	2	3	4	5	6	7	8	9	10+
Documents	23	47	73	123	136	145	161	111	83	61	42

In order to evaluate the accuracy and validity of the thematic categorization results, a precision-recall diagram has been constructed. In information retrieval (IR), precision is defined as the number of retrieved relevant items over the number of total retrieved items. Recall is defined as the number of retrieved relevant items over the total number of relevant items:

$$p = precision = \frac{relevant\ retrieved\ items}{retrieved\ items} \tag{1.24}$$

$$r = recall = \frac{relevant\ retrieved\ items}{relevant\ items} \tag{1.25}$$

In our case, of course, "retrieval" refers to thematic categorization. Alternative terminologies are also widely used in classification problems, e.g. [26]. The performance for an "deal" system is to have both high precision and recall. Unfortunately, these are conflicting entities and cannot be at high values at the same time. Therefore, instead of using a single value of precision and recall, a Precision-Recall (PR) graph is typically used to characterize the performance of an IR system.

In order to acquire multiple pairs of precision and recall and draw the diagram, different thresholds have been employed on the degree of relevance of a document to a thematic category, i.e., threshold T_c was let vary from 0.3 to 0.9. Binary labels were manually assigned to all 1005 documents for five thematic categories (sports, military, arts, education and entertainment), in order to construct a ground-truth to be used for comparisons to the results of thematic categorization and subsequent measurement of precision and recall. The resulting diagram is presented in Figure 1.9 in red.

The yellow line of the same figure presents the precision-recall diagram for the Faethon system when the same five thematic categories are used as queries and thematic categorization information is not used in the query processing. We note that the Faethon query processing scheme also takes advantage of the knowledge stored in the encyclopedia and considers the context in query interpretation, query expansion and index matching operations [2]. Thus, any difference in the two diagrams reflects the operation of the proposed thematic categorization algorithm. We can see that for similar values of recall the thematic categorization has higher precision values, as it does not include documents that contain related words but are not truly related to the thematic category.

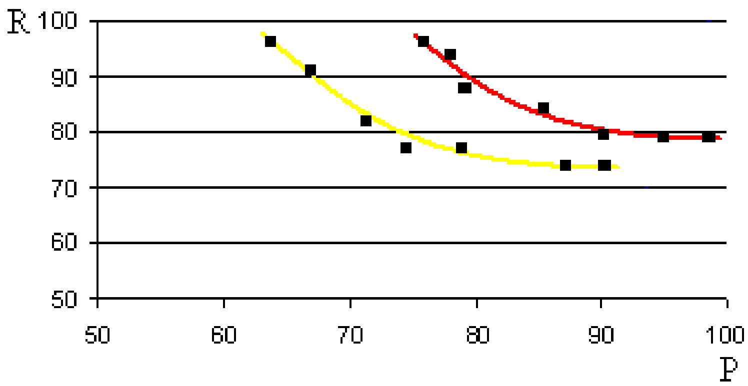

Fig. 1.9. Precision-Recall Diagram

1.6 Conclusions

The semantic gap refers to the inability to efficiently match document semantics with user semantics, mainly because neither is usually readily available in a useful form. In this work we have made an attempt to extract the former, relying on fuzzy algebra and a knowledge base of fuzzy semantic relations.

Specifically, we started by describing the construction of a semantic index using a hybrid approach of processing and manual annotation of raw multimedia information. We then explained how the index can be analyzed for the detection of the topics that are related to each multimedia document. The existence of noise and uncertainty in the semantic index has also been considered. Our approach is based on the notion of context and the utilization of fuzzy taxonomic relations.

As multimedia content is becoming a major part of more and more applications every day, the applications of this work are numerous. As more important, one may mention automated multimedia content organization, indexing and retrieval, usage history analysis, user adaptation, efficient multimedia content filtering and semantic unification of diverse audiovisual archives [5].

Although this work is contributing in the direction of bridging the semantic gap, a lot more has to be done before one may claim that the problem is solved. In this chapter we have assumed the existence of a semantic index, which cannot yet be constructed in an automated manner. Major focus in our future work will be given to the automated mapping of MPEG-7 syntactically described objects and events to their corresponding semantic entities, based on techniques such as graph matching. Another area of future research is the selection of optimal fuzzy operators for the most meaningful semantic output. Our findings so far indicate that this selection is not independent from the knowledge itself. Finally, one more direction is the utilization of existing crisp taxonomies for the generation of the knowledge that is required for the analysis of the multimedia documents.

References

1. Akrivas G., Stamou G., Fuzzy Semantic Association of Audiovisual Document Descriptions, *Proceedings of International Workshop on Very Low Bitrate Video Coding (VLBV)*, Athens, Greece, Oct. 2001.
2. Akrivas G., Wallace M., Andreou G., Stamou G. and Kollias S., Context - Sensitive Semantic Query Expansion, *Proceedings of the IEEE International Conference on Artificial Intelligence Systems (ICAIS)*, Divnomorskoe, Russia, September 2002.
3. Akrivas G., Wallace M., Stamou G. and Kollias S., Context - Sensitive Query Expansion Based on Fuzzy Clustering of Index Terms, *Proceedings of the Fifth International Conference on Flexible Query Answering Systems (FQAS)*, Copenhagen, Denmark, October 2002.

4. Akrivas G., Ioannou S., Karakoulakis E., Karpouzis K., Avrithis Y., Delopoulos A., Kollias S., Varlamis I. and Vaziriannis M., An Intelligent System for Retrieval and Mining of Audiovisual Material Based on the MPEG-7 Description Schemes, *Proceedings of European Symposium on Intelligent Technologies, Hybrid Systems and their Implementation on Smart Adaptive Systems (EUNITE 01)*, Tenerife, Spain, December 12-14, 2001.
5. Avrithis Y. and Stamou G., FAETHON: Unified Intelligent Access to Heterogenous Audiovisual Content, *Proceedings of the International Workshop on Very Low Bitrate Video Coding (VLBV)*, Athens, Greece, Oct. 2001.
6. Avrithis Y., Stamou G., Delopoulos A. and Kollias S., Intelligent Semantic Access to Audiovisual Content, *Lecture Notes in Artificial Intelligence*, Springer-Verlag, Vol. 2308, 2002, pp. 215-224.
7. Bowman C.M., Danzing P.B., Manber U. and Schwartz F., Scalable Internet resources discovery: research problems and approaches, *Communications of the ACM*, Vol. 37, pages 98-107, 1994.
8. Chan S.S.M., Qing L., Wu Y. and Zhuang Y., Accommodating hybrid retrieval in a comprehensive video database management system. *IEEE Trans. on Multimedia*, 4(2):146-159, June 2002.
9. Chen W. and Chang S.-F., VISMap: an interactive image/video retrieval system using visualization and concept maps. *In Proc. IEEE Int. Conf. on Image Processing*, volume 3, pages 588-591, 2001.
10. Ciocca G. and Schettini R., A relevance feedback mechanism for content – based image retrieval, *Information Processing and Management*, Vol. 35, pages 605-632, 1999.
11. Del Bimbo, A. *Visual Image Retrieval*, CA: Morgan Kaufmann, San Francisco, 1999.
12. Dubois D. and Prade H., The three semantics of fuzzy sets, *Fuzzy Sets and Systems*, Vol. 90,pages 142-150, 1997.
13. Zhao R. and W.I. Grosky, Narrowing the Semantic Gap-Improved Text-Based Web Document Retrieval Using Visual Features, *IEEE Trans. on Multimedia*, Special Issue on Multimedia Database, Vol. 4, No 2, June 2002.
14. Klir G. and Bo Yuan, *Fuzzy Sets and Fuzzy Logic, Theory and Applications*, New Jersey, Prentice Hall, 1995
15. Kraft D.H., Bordogna G., Passi G., Information Retrieval Systems: Where is the Fuzz?, *Proceedings of IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Anchorage, Alaska, May 1998
16. Maedche A., Motik B., Silva N. and Volz R., MAFRA - An Ontology MApping FRAmework in the Context of the SemanticWeb, *Proceedings of the Workshop on Ontology Transformation ECAI2002*, Lyon, France July 2002.
17. Miyamoto S., *Fuzzy Sets in Information Retrieval and Cluster Analysis*, Kluwer Academic Publishers, Dordrecht / Boston / London 1990.
18. Salembier P. and Smith J. R., MPEG-7 Multimedia description schemes, *IEEE Transactions on Circuits and Systems for Video Technology, 11(6):748-759*, Jun 2001.
19. Salton G. and McGill M.J., *Introduction to Modern Information Retrieval*, McGraw-Hill, New York, 1983.
20. Santini, S., *Exploratory Image Databases: Content-based Retrieval*, New York: Academic, 2001.
21. Theodoridis S. and Koutroumbas K., *Pattern Recognition*, Academic Press, 1998.

22. Tsechpenakis G., Akrivas G., Andreou G., Stamou G. and Kollias S., Knowledge - Assisted Video Analysis and Object Detection, *Proceedings of European Symposium on Intelligent Technologies, Hybrid Systems and their Implementation on Smart Adaptive Systems* Albufeira, Portugal, September 2002.
23. Wallace M., Athanasiadis T., Avrithis Y., Stamou G. and Kollias, S., A mediator system for hetero-lingual audiovisual content, *Proceedings of the International Conference on Multi-platform e-Publishing*, November 2004, Athens, Greece.
24. Wallace M., Avrithis Y., Stamou G. and Kollias S., Knowledge-based Multimedia Content Indexing and Retrieval, in Stamou G., Kollias S. (Editors), Multimedia Content and Semantic Web: Methods, Standards and Tools, Wiley, 2005.
25. Wallace M., Karpouzis K., Stamou G., Moschovitis G., Kollias S. and Schizas C., The Electronic Road: Personalised Content Browsing, *IEEE Multimedia 10(4)*, pp. 49-59, 2003
26. Yang Y., An Evaluation of Statistical Approaches to Text Categorization, *Journal of Information Retrieval*, Vol 1, No. 1/2, pp 67–88, 1999.
27. ISO/IEC JTC 1/SC 29 M4242, "Text of 15938-5 FDIS Information Technology – Multimedia Content Description Interface – Part 5 Multimedia Description Schemes," October 2001.