# The Effectiveness of Concept Based Search for Video Retrieval

**Claudia Hauff** and **Robin Aly** and **Djoerd Hiemstra**
Computer Science
University Twente
P.O. Box 217
7500 AE Enschede
The Netherlands
e-mail: {c.hauff, r.aly, d.hiemstra}@ewi.utwente.nl

## Abstract

In this paper we investigate how a small number of high-level concepts derived for video shots, such as *Sports*, *Face*, *Indoor*, etc., can be used effectively for ad hoc search in video material. We will answer the following questions: 1) Can we automatically construct concept queries from ordinary text queries? 2) What is the best way to combine evidence from single concept detectors into final search results? We evaluated algorithms for automatic concept query formulation using WordNet based concept extraction, and we evaluated algorithms for fast, on-line combination of concepts. Experimental results on data from the TREC Video 2005 workshop and 25 test users show the following. 1) Automatic query formulation through WordNet based concept extraction can achieve comparable results to user created query concepts and 2) Combination methods that take neighboring shots into account outperform more simple combination methods.

## 1 Introduction

Bridging the semantic gap is a key problem in multimedia information retrieval [Sebe, 2003]. This gap exists between the well understood extraction methods of low level features from media files (e.g. color histograms or audio signals) and the high level concepts users express their information needs with (e.g. *Find me pictures of a sunrise*). This problem applies especially to video retrieval where the detection of the semantic concepts has become a research focus in recent years [Naphade and Smith, 2004]. This paper investigates the problem of identifying the relevant concepts given the user's text query, and it investigates how multiple concepts need to be combined to retrieve the best video shots on data from the TREC video (TRECVID) search task of 2005.

The goal of the TRECVID search task is the retrieval of video shots which are relevant to the user. We adopt the definition of a semantic concept from Snoek at al. [Snoek *et al.*, 2006b] where a concept is defined as something which must appear clearly in the static key frame of the video shot to return. Thus the expression does not cover concepts which are only represented in the audio content or more abstract concepts such as *World Peace*.

The generally used approach to detect concepts in video data is to train several so called detectors through positive and negative examples in order to recognize the appearance of a concept. The problem of how to determine the set of required detectors applies even for limited domains. The most commonly used metaphor for this problem is to define an infinite *semantic space*. The objective is to create detectors for a certain set of concepts which should allow to answer all possible queries [Naphade *et al.*, 2005]. Besides the issue of selecting appropriate concepts for a particular domain, another question is how to handle requests for concepts which are not directly present in the set of available concepts or require the usage of more then one concept. For example, a user might search for *Condoleezza Rice* but the search system only has the concepts *Face* and *Women* available. Due to the lack of knowledge about the structure of the *semantic space*, it is not an option to simply increase the number of detectors up to the point where all requested concepts are covered. Thus, some concepts have to be expressed as a combination of concepts for which detectors exist. This problem has not been satisfactory addressed yet [Snoek *et al.*, 2006a].

Users searching an image or video collection cannot be expected to know the concepts that have been used in the concept detection step. User queries usually either consist of a few keywords (e.g. *Beach*) or more elaborate natural language requests (e.g. *Find me pictures of a beach with people.*). In the best case the query contains one or more of the concept names and syntactic matching is sufficient, however often this will not be the case (for instance, in TRECVID available concepts include *Outdoor*, *Waterscape* and *People* but not beach). Hence, the first task is the extraction of the concepts underlying the queries. The query concepts and the concepts available for the collection are then matched and a ranking of relevant concepts is derived that shall resemble the information need expressed in the query as closely as possible . When viewing the relevant concepts analogously to relevant documents in the information retrieval setting, the quality of concept extraction can be evaluated with information retrieval methods. In order to create relevance judgments we performed a user study and evaluated our automatic concept query formulation algorithms against the queries formulated by the users. This has the advantage that automatic query formulation can be evaluated independently.

In previous work [Aly *et al.*, 2007], we evaluated approaches to combing two concepts to form a new composite concept. In this paper we extend this work to the combination of more than two concepts. Prior to this we introduce a framework of formulas which simplifies the process of building new scoring formulas. Given the ranked list of concepts from our approaches above there is still the open question of which of those to choose for the actual combination. We introduce a number of mechanisms to solve this task.

The rest of this paper is organized as follows: In Section 2 we briefly give an overview of related work. Section 3 describes the methods utilized for mapping user queries to a ranked list of concepts. In Section 4 the scoring of composite concepts is described, and ways to choose concepts from a ranked list are evaluated. The following section describes the experiments performed to evaluate to presented methods (Section 5). Finally, Section 6 concludes and proposes future work.

## 2 Related Work

A lot of work on concept detection has been done in the context of the TRECVID Workshop [Smeaton *et al.*, 2006]. The search task in TRECVID requires the participants to return for each topic the first 1000 entries of a ranked list of shots. Each topic consists of a textual part and example multimedia material. Together with the main raw collection the participants get the results of the high-level feature extraction task and speech transcripts from an Automatic Speech Recognition System (ASR). We use the data of TRECVID 2005 to evaluate our methods.

The query concept extraction process is aided by background knowledge in the form of thesauri or more general ontologies. These are hierarchical and associative structures usually created manually by experts that capture domain-independent or domain-dependent knowledge. The most widely used knowledge base for the general domain today is WordNet [Fellbaum, 1998], a semantic dictionary whose content expresses common-sense world knowledge which is also a popular knowledge source for TRECVID participants. WordNet is applied in two directions: expansion of queries, documents and concept descriptions and the determination of relatedness scores between concepts. In [Koskela *et al.*, 2006; Sjóberg *et al.*, 2006] the given concepts were located in WordNet and the concept description was expanded by the concept names' synonymous terms. The topics were then syntactically matched against the concept descriptions. A more general approach was adopted in [Campbell *et al.*, 2006], namely the Adapted Lesk algorithm [Banerjee and Pedersen, 2002]. It also relies on the overlap between topic text and the concept descriptions, but furthermore takes advantage of WordNet's graph structure and considers related concepts and their descriptions as well. Instead of matching queries and concepts based on their descriptions, their relatedness can also be measured directly on WordNet's graph structure. [Snoek *et al.*, 2006a] link all possible query nouns and the concepts to entries in WordNet and determine their relatedness by applying Resnik's Information-based algorithm [Resnik, 1995]: the relatedness between two concepts equals the information content of their most specific common parent.

Exploiting the relationship between concepts is related to the creation detectors for combined concepts. The idea behind the multi-concept relationships is to let the scoring process for a concept be influenced by the relationship with other related concepts, for example the likelihood of observing the concept *Bus* decreases when observing an *Indoor* setting with a high score. Rong Yan [Yan, 2006] give a good overview of the available techniques to do this. The link to our proposed method is that the multi-concept relationship approach tries to improve detectors by considering the presence of related concepts and the presented approach creates new concepts. Thus both consider multiple concepts.

The MediaMill Group [Snoek *et al.*, 2006b] evaluated several ways of combining *low-level* features, namely color-histograms and associated text generated by performing ASR, into high-level concept detectors. Each strategy is based on a vector of a number of low-level features. The detector relies on support vector machines (SVM) [Vapik, 1998] and is trained on designated training data in order to accurately assign scores to shots from other data sources. In their experiments, they investigated different types of low-level features and their respective impacts: 1) video features only, 2) associated text only, 3) video features and associated text (early fusion), 4) a combination of the output of 1) and 2) (late fusion) and finally 5) a combination of the output of methods 1)-4). For TRECVID 2005 they trained and evaluated 101 concept detectors on approximately 30,000 shots. On average method 1) was performing the best on the TRECVID dataset. Hence textual features were not particularly beneficial. The reason for this was not explicitly researched. It seems plausible that the data was not suitable for speech recognition - because ASR and machine translation from Chinese and Arabic speech introduced too much noise. The output of their set of concept detectors are rankings for the search data together with the ground-truth and rankings on the test dataset that was used for the the high level feature extraction evaluation. We use the scores of the 101 concept detectors on the search data to verify our ideas and employ the results from the test data to judge the quality of a detector.

## 3 Concept Extraction

### 3.1 WordNet

WordNet [Fellbaum, 1998] is an online lexical database developed at Princeton University that was inspired by psycholinguistic theories. It is continuously enlarged and updated by human experts and as already been pointed out can be viewed as a general domain knowledge base. WordNet's building blocks are sets of synonymous terms[1] (so-called *synsets*) each representing one lexical concept that are connected to each other through a range of semantic relationships. Relations between terms instead of synsets exist as well but are not very frequent.

A small part of WordNet is shown as a graph in Figure 1: the synsets are represented by nodes and an edge exists between two synsets if they are semantically related. A synset can consist of several terms and each term is contained in $m$ synsets with $m$ being the number of senses the term has (identified by the sense numbers $\#X$ in Figure 1).

Relationships exist only between synsets of the same word type, hence there are separate structures for nouns, verbs, adjectives and adverbs with nouns make up the largest fraction of WordNet. We restrict ourselves to a short overview of the relations that we utilized in our approach. In the following paragraphs, $s_1$ and $s_2$ represent synsets and $t_1$ and $t_2$ represent terms.

**Hypernymy, hyponymy** (nouns): $s_1$ is a *hypernym* of $s_2$ and $s_2$ is a *hyponym* of $s_1$ if $s_1$'s meaning contains $s_2$'s, e.g. {*vessel, watercraft*} is a hypernym of {*ship*}.

**Hypernymy, troponymy** (verbs): $s_1$ is a *hypernym* of $s_2$ and $s_2$ is a *troponym* of $s_1$ if $s_2$ is a certain manner of $s_1$ e.g. {*walk*} is a hypernym of {*stroll, saunter*}.

[1] A term can be a single word, a compound or a phrase.

**Holonymy, meronymy** (nouns): $s_1$ is a *holonym* of $s_2$ and $s_2$ is a *meronym* of $s_1$ if $s_2$ is a member of or part of $s_1$, e.g. {*fleet*} is a holonym of {*ship*}.

**Sibling** (nouns, verbs): $s_1$ and $s_2$ are *siblings* if they share a direct hypernym e.g. {*ship*} and {*yacht, racing yacht*} are siblings as they share the hypernym {*vessel, watercraft*}.

**Entailment** (verbs): $s_1$ *entails* $s_2$ if $s_1$ implies $s_2$ e.g. {*buy, purchase*} entails {*pay*}.

**Verb group** (verbs): $s_1$ and $s_2$ belong to the same *verb group* if they have a similar meaning (manually grouped by human experts)

**Derivationally related form** (nouns, verbs): the noun $t_1$ has a derivationally related noun (or verb) form $t_2$ if they are morphologically related and semantically linked e.g. the noun *machine* and the noun *machinist* or the verb *cook* and the noun *cook*.

**Similar to** (adjectives): $s_1$ and $s_2$ are similar to each other if one is more general than the other e.g. {*yellow, yellowish, xanthous*} and {*chromatic*} are similar.

Furthermore, WordNet also provides glosses for all synsets, which consist of definitions or sentences that show the synset's usage. Examples are here the gloss "a craft designed for water transportation" for {*vessel, watercraft*} and the gloss "of the color intermediate between green and orange in the color spectrum; of something resembling the color of an egg yolk" for {*yellow, yellowish, xanthous*}.

### 3.2 Extracting Concepts from Queries

In order to determine the ranking of concepts that best describes a query, the relatedness scores between the concepts and the query need to be determined. This can be done on two levels: on the synset level or on the term level. A number of algorithms have been proposed [Banerjee and Pedersen, 2002; Patwardhan and Pedersen, 2006; Resnik, 1995; Jiang and Conrath, 1997; Lin, 1998; Lea, 1998; Wu and Palmer, 1994] that differ in what part of WordNet they utilize - glosses, synset terms and various relationship types with different weighting schemes. In the next two sections, the approaches chosen for our experiments are explained in greater detail. First, a term-level gloss-based approach is presented, then three synset-level graph-based approaches are introduced. In both cases, it is assumed that the concepts have been (manually) linked to the correct corresponding synsets in WordNet.

**Using WordNet's glosses**

This approach does not require extensive preprocessing of the queries as the graph-based approaches do. Furthermore, it is not restricted by word types: if a query noun is found in a gloss of a verb for example, the verb concept is deemed likely to be relevant. Graph-based approaches on the other hand usually cannot cross part-of-speech boundaries. The gloss of a concept's synset as well as the glosses of related synsets are used to create a *concept document*. The type of relations used, the maximum depth and the glosses' weightings are freely settable parameters. A depth of 0 means that only the gloss of the synset itself is added to the concept document, a depth of 1 includes the directly related synsets as well, etc. Possible weighting schemes include uniform weighting of every gloss and linear weighting, which linearly decreases the weight of the glosses the larger the depth. The concept documents are then treated as a document collection and keyword-based text retrieval is performed - the document ranking corresponds to the concept ranking.

> head of state chief of state the chief public representative of a country who may also be the head of government chancellor premier prime minister the person who is head of state (in several countries) Prime Minister PM premier the person who holds the position of head of the government in England president the chief executive of a republic President of the United States United States President President Chief Executive the person who holds the office of head of state of the United States government; "the President likes to jog every morning" sovereign crowned head monarch a nation's ruler or head of state usually by hereditary right

Figure 2: A concept document for the concept *government leader* which was mapped to WordNet's {*head of state, chief of state*} synset. Hyponym-related synset glosses up to a depth of 1 were added.

An example of a concept document is shown in Figure 2. It also demonstrates one of WordNet's drawbacks: WordNet is updated and altered by human experts, hence inevitably there will be a bias in what concepts make it into WordNet and what concepts do not. While the Prime Minister of the United Kingdom and the President of the United States occur as concepts, the heads of almost all other countries cannot be found.

**Using WordNet's Graph Structure**

Determining the semantic relatedness scores requires a number of preprocessing steps as depicted in Figure 3. First of all, the word types of the query terms need to be found with a part-of-speech (POS) tagger. Since most terms have more than one sense their meaning in this particular context needs to be determined. This step is called word sense disambiguation (WSD) and can also utilize WordNet. In the simplest case the most common sense (which is provided by WordNet) is chosen.

Having located the query terms' corresponding WordNet synsets makes it possible to use graph theoretic measurements to determine the semantic relatedness between the query concepts and the given concepts. A very simple measure is the hierarchical shortest path measurement $rel_{HS}(s_1, s_2)$: how many hypernymy/hyponymy edges[2] $len$ of the WordNet graph need at least to be traversed to reach a synset $s_2$ from synset $s_1$? There are problems though, as WordNet is a small-world network [Sigman and Cecchi, 2002], hence within the connected part of the graph two nodes can always be reached within a few steps. Another issue is that at the synsets close the root node are quite dissimilar from each other (e.g. {*object, physical object*} is a direct hypernym of emph{ice}) whereas deep in the hierarchy they tend to be very similar (e.g. {*cab, hack, taxi, taxicab*} is a direct hypernym of {*minicab*}). For

[2]If only the hypernymy/hyponymy relationship is utilized, the noun graph becomes hierarchical and the measures are often called *semantic similarity* instead of the more general *semantic relatedness* which considers all types of relationships.

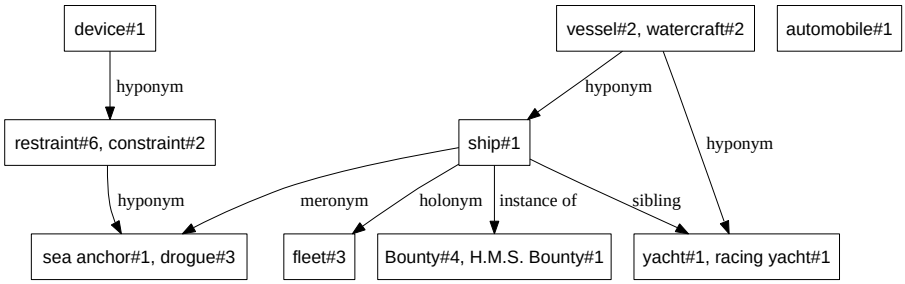


Figure 1: A part of WordNet's noun graph

these reasons, measurements usually prohibit edge walks along certain relationship types [Hir, 1998], they include information about WordNet's depth [Lea, 1998; Wu and Palmer, 1994] or exploit information drawn from analysing large corpora [Resnik, 1995; Jiang and Conrath, 1997; Lin, 1998]. An overview of five WordNet-based relatedness measurements and their performances is given in [Budanitsky and Hirst, 2006].

[Lea, 1998] determine the relatedness score of two synsets also solely based on the hypernymy/hyponymy relationship. In contrast to $rel_{HS}$, the number of edges between $s_1$ and $s_2$ is scaled by the maximum depth of the WordNet hierarchy.

$$rel_{LC}(s_1, s_2) = -\log \frac{len(s_1, s_2)}{2 \times \max\limits_{s \in WordNet}[depth(s)]} \quad (1)$$

[Wu and Palmer, 1994] exploit not the global depth of WordNet but instead the depth of the *lowest super-ordinate* ($lso$) of the two synsets, that is the most specific synset that subsumes both synsets (e.g. the lowest super-ordinate for {*ship*} and {*yacht, racing yacht*} is their common hypernym {*vessel, watercraft*}). Let $z = lso(s_1, s_2)$, then the relatedness is given by

$$rel_{WP}(s_1, s_2) = \frac{2 \times depth(z))}{len(s_1, z) + len(s_2, z) + 2 \times depth(z)} \quad (2)$$

# 4 Concept Combination for Search

This section studies the possibilities to combine multiple concepts in video retrieval. Our previous studies revealed that the combination of two concepts improves performance. We believe the extension to multiple concepts is beneficial because many concepts stand in a inherent *leads directly to* relationship to others. For example, the correct detection of the concept *Face* directly leads to the presence of the the concept *Person*. Thus, if searching for *A person in the street* the search for the concepts *Person Face Street Outdoor* will be beneficial in case we have a good *Face* and *Outdoor* detector. This, of course, assumes that persons are mainly shown with their face into the camera and that all streets are outdoor.

The rest of this section proceeds as follows: first the basic operations used are introduced (Section 4.1), followed by an overview of the new ranking formulas tested (Section 4.2). In Section 4.3 methods on how to identify the concepts to use in a query, given the list of concepts selected by the concept extraction approach presented in the previous section.

$$\chi(c, s_j) = \text{score } s_j \text{ using } c \quad (3)$$
$$\psi(C, s_j) = \text{score } s_j \text{ using } C \quad (4)$$

$\chi$ Functions:

$$r(c, s_j) = \text{original score} \quad (5)$$
$$factor(c, s_j) = \log(r(c, s_j)) \quad (6)$$
$$smooth(c, s_j) = \frac{\sum_{i=j-nh}^{j+nh} \delta(|i-j|) r(c, s_i)}{\sum_{i=j-nh}^{j+nh} \delta(|i-j|)} \quad (7)$$
$$weighted(c, s_j) = ap(c) \cdot r(c, s_j) \quad (8)$$

Combination Functions:

$$sum(\chi, C, s_j) = \frac{\sum_{c \in C} \chi(c, s_j)}{|C|} \quad (9)$$
$$sumC(\chi, \psi, C, s_j) = \sum_{c \in C} \chi(c, s_j) \frac{\psi(C \setminus c, s_j)}{|C| - 1} \quad (10)$$

Figure 4: Basic Functions

## 4.1 Basic Operations for Ranking Functions

We refined the list of scoring functions and extended them to handle more then two base concept detectors. Formally their task is to calculate the score for a shot $s_j$ based on the detectors of a set of concepts $C$. We identified two different classes of basic operations: 1) functions which only use one concept $c$ for their score calculation $\chi$ and 2) combination functions which operate on a set of concepts $\psi$.

$r(c, s_j)$ (5) simply returns the score of the shot $s_j$ as calculated by the detector for concept $c$. Instead of introducing a combination function that sums the scores of two concept ranking functions, and another that multiplies them, we define a function $factor(c, s_j)$ (6). Summed logarithmic scores produce the same ordering of shots as multiplied original scores. Using the function $factor$ is beneficial due to less numerical precision loss in case of a multiplication. Another reason is to keep the set of combination operations small.

The function $smooth$ (7) assumes that it is more likely that a concept $c$ appears in the shot $s_j$ if it also appears in previous or following shots. Similar approaches have been investigated using the text from automatic speech recognition associated with shots [Hauptmann *et al.*, 2006]. We define the surrounding neighborhood as a fixed number $nh$ of shots before and after the actual shot $s_j$ that contribute to the score of $s_j$. We expect that shots which are further away from $s_j$ to rank, to have less influence on the likelihood of the presence of concept $c$. We model this fact in

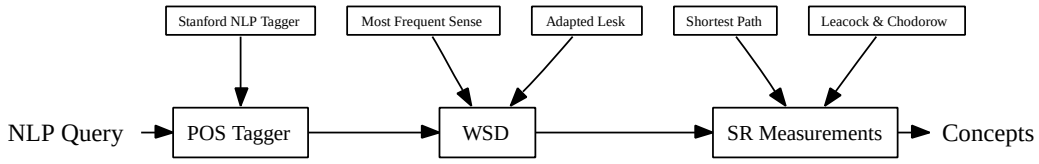


Figure 3: Converting queries to concepts.

$$add(C, s_j) = sum(r, C, s_j) \quad (11)$$
$$cbw(C, s_j) = sum(weighted, C, s_j) \quad (12)$$
$$mult(C, s_j) = exp(sum(factor, C, s_j) \quad (13)$$
$$n(C, s_j) = sumC(r, sum(smooth), C, s_j) \quad (14)$$
$$\psi_{log} = log(sum(smooth)) \quad (15)$$
$$\psi_{logs} = log(sum(smooth)) \quad (16)$$
$$nm(C, s_j) = exp(sumC(factor, \psi_{log}, C, s_j)) \quad (17)$$
$$nw(C, s_j) = sum(weighted, \psi_{logs}, C, s_j) \quad (18)$$
$$sa(C, s_j) = sum(smooth, C, s_j) \quad (19)$$
$$sm(C, s_j) = exp(sum(\psi_{log}, C, s_j)) \quad (20)$$

Figure 5: Concrete Functions

a weighting function $\delta$ which takes the absolute distance $[0 \dots nh]$ of a shot as an argument and returns a weight in the interval $[0 \dots 1]$ to weight the score of the shot. We define the smooth function as follows:

Let $\delta(x)$ be a function of the distance from the shot that determines the influence of the neighboring shots. We created three alternatives for $\delta$. The first version $\delta_{constant}(x) = 1$ weights all shots uniformly. This will serve as a base line. The function $\delta_{linear}(x) = 1 - \frac{x}{nh+1}$ lowers the weight of a shot linearly in its distance to $s_j$. The furthest shots on both sides will still have a small positive weight. The version $\delta_{exp}(x) = -exp(-x)$ lowers the weight of shots in an exponential fashion.

As ranking functions differ in their precision of detecting their base concepts [Snoek *et al.*, 2006b] we created a possibility to weight their output. $weighted(c, s_j)$ (8) weights the outcome of the ranking function for concept $c$ by the average precision $ap(c)$ achieved for this concept on the test dataset.

We identified two basic combination methods. The first, $sum(\chi, C, s_j)$ (9) takes as $\chi$ a function which should be executed for each particular concept. $C$ is the set of concepts which it should perform the function for. It then sums up the results from execution of $\chi$ on each concept from $C$. To keep the scores within $[0..1]$ it divides the result from the sum by the number of concepts. The function $sumC(\chi, \psi, C, s_j)$ (10) allows each summand to be calculated from two parts: A score from a function using the current concept on the shot (Class $\chi$) and a function which operates on all other concepts in the set passed to the function $\psi$. In order to assure range intervals the output of this function is divided by the number of concepts the calculation is build on. The rational is that the score for a concept on a certain shot could be influenced by the performance of other ranking functions.

### 4.2 Ranking With Combined Concepts

Based on these basic operations we extended methods we already studied in [Aly *et al.*, 2007] for the use of two concepts. The function $add(C, s_j)$ (11) is a simple summation of the base scores. The derived version $cbw(C, s_j)$ (12) weights the summands by their AP in the test set. In the function $mult(C, s_j)$ (13) first the sum of the logarithms out of each base score is calculated. At the end a $exp()$ function is applied to get the scores again in the interval $[0..1]$.

The Neighbor function $n(C, s_j)$ (14) considers all base scores multiplied with the average of the smoothed scores of the other concepts. $nm(C, s_j)$ (17) is an extension of the $mult$ function which is weighting the individual scores by the $log()$ of averaged smoothed scores of other concepts.

As described above it is the case that some concept detectors are less precise then others. Therefore we create versions of the $n$ function, namely $nw(C, s_j)$ (18) which additionally weights the score of the individual scoring function by the AP of the detector in the test data.

A new class of scoring functions are the functions which only operate on smoothed values. The $sa(C, s_j)$ (19) takes the average of all smoothed scores. The function $sm(C, s_j)$ (20) does the equivalent but with the described method to effectively multiply summands.

### 4.3 Selection of Concepts

The input for the concept combination algorithms are ordered lists of concepts. The problem now is what concepts to employ during the ranking. The most obvious is of course to combine the whole list of concepts. However there are a lot of concepts which were only chosen once or have very little effect on the search performance. Out concept extraction method could return all available available concepts, in which case some of them will definitely have negative impact for search performance. To overcome this problem we use a Top-N approach to only select the first $n$ concepts of a list.

## 5 Evaluation

### 5.1 Concept Extraction

Not every concept could be attached to exactly one synset in WordNet, some concepts were linked to several synsets. The concept *natural disaster* for instance does not occur as such in WordNet 2.1 but instead was represented by the synsets {*flood, inundation, deluge, alluvion*}, {*earthquake, temblor, seism*}, {*storm, violent storm*} and {*volcanism*}. Another problem arose for several person concepts like *A. Sharon* or *E. Lahoud* which have no representation in WordNet. In those cases, the concepts were added to WordNet as instances of an appropriate synset such as {*head of state, chief of state*} and their gloss consists of the concept name alone.

| Run | Depth | MAP | P@5 |
|---|---|---|---|
| uniform/sibling | 0 | 0.268 | 0.246 |
| uniform/noSibling | 4 | 0.296 | 0.225 |
| linear/sibling | 4 | 0.286 | 0.200 |
| linear/noSibling | 4 | 0.297 | 0.225 |

Table 1: Results for the gloss-based approach. 4 types of runs were performed with varying depth: uniform weighting with siblings, uniform weighting without siblings, linear weighting with siblings and linear weighting without siblings. The best performing run for each type in terms of P@5 is shown together with its depth parameter.

f

**Golden Standard**

In order to evaluate the different concept extraction algorithms separately from the concept combination part, we developed a *golden standard* for the TRECVID 2005 topics. 25 users were given the topics and asked to return those concepts of the 101 available concepts that best describe the information need expressed in the topics. No restriction was given on the number of concepts the users could choose. On average users chose 6.96 concepts per topic. The spread between the users was quite large: the lowest average was 4.42, the maximum average was 10.67. Furthermore, for many topics the agreement between the users was surprisingly low. 11 of the 25 topics had more than 20 different concepts returned at least once. We derived a concept ranking for each topic from this survey by considering all concepts of a topic that more than 50% of the participants had chosen and ranked them accordingly. The average number of concepts per topic was reduced 3.2, the minimum number of concepts is 2 (for topics 155, 157, 164, 166, 170) and the maximum number is 7 (for topic 159). As mentioned before, viewing the relevant concepts as relevant documents in the information retrieval setting, allows us to evaluate our approaches with information retrieval performance measures, namely mean average precision (MAP) and precision at 5 documents (P@5).

**Gloss-Based Approach**

For the gloss-based conversion, we utilized the *hyponymy*, *meronymy*, *entailment*, *sibling*, *verb group*, *derivationally related form* and *similar to* relationships described in Section 3.1. The depth was varied between 0 and 5 and the uniform and linear weighting schemes were tested. Finally due to the large volume of siblings for a number of concepts we also considered the influence they have and run the algorithms with and without the inclusion of this particular relationship. For the retrieval experiments we employed the Lemur Toolkit for Information Retrieval[3]. The documents and topics were stemmed and stopwords were removed. The language modeling approach with Jelinek-Mercer smoothing was used for retrieval purposes. We report the results in Table 1. Since the concept combination step relies on the Top-N ranked concepts, P@5 was deemed a more important measure than MAP.

Surprisingly, using the synset glosses without adding related concepts performs best for P@5. The sibling relationship hurts the performance across all runs tested except for one. The weighting scheme does not have a large influence on the results.

[3]http://www.lemurproject.org/

| Run | MAP | P@5 |
|---|---|---|
| $rel_{HS}$ | 0.370 | 0.217 |
| $rel_{LC}$ | 0.366 | 0.217 |
| $rel_{WU}$ | 0.345 | 0.200 |

Table 2: Results for the graph-based approaches.

**Graph-Based Approaches**

POS tagging the queries was performed with the Stanford NLP Tagger[4]. The relatedness measures $rel_{HS}$, $rel_{LC}$ and $rel_{WU}$ introduced in Section 3.2 were investigated in two variants: 1) word sense disambiguation of the query concepts was reduced to choosing the most common sense and 2) the query terms $q_i$ were tested with all their senses against the concepts and the maximum relatedness score was returned:

$$rel(q_i, s_j) = \underset{q_i \in s_k}{max}[rel(s_k, s_j)]. \quad (21)$$

The differences between 1) and 2) proved to be small, in Table 5.1 the results of the most common sense approach are presented. While $MAP$ considerably increases over the gloss-based approach, $P@5$ is harmed. Thus, for the TRECVID 2005 topics, using only the glosses of the synsets corresponding to the concepts is the best approach among all tested ones.

## 5.2 Combined Concept Search

Most of our combinations methods depend on the $smooth$ method. There are two free parameters which will affect the performance: The degrading function $\delta$ and the size of the neighborhood $nh$. We first evaluate the best parameter setting performance in order to justify which of the combinations to employ in the later combination. First we evaluated which $\delta$ function was the best. We did this by taking the average AP of the first top-n concepts for each query with this $\delta$. We did this for top-n following $\{2, 4, 6, 8\}$. The results are shown in Table 3 (a).

$\delta_{const}$ performs for all top-n values best, thus it is used to evaluate the variations of $nh$. Results with other $\delta$ functions are not shown due to space limitations but did not yield other conclusion. Table 3 (b) shows the MAPs of the $r$ and the $smooth$ function with $\delta_{const}$ for $nh \in \{2, 5, 8\}$. We stopped at $nh = 8$ because it was the first drop in the MAP. As we deem it to be realistic that this will not improve with higher $nh$ we limit ourselves to this sample. The setting $nh = 2$ is always worse than the other results. The setting $nh = 5$ improves MAP compared to $r$ by 24% in average. With $nh = 8$ the improvement is a bit lower with 18%. This brings us to the conclusion that we use for our combination functions $smooth$ with the parameter setting $\delta = \delta_{const}$ and $nh = 5$.

After evaluation of the $smooth$ method we use the best settings to test the concept combination methods. We evaluated them against the Golden Standard and our concept extraction methods. The parameter $N$, the Top-$N$ extracted concepts was tested in a range from $2..8$. It turned out that Top-2 showed the best MAPs. Table 4 shows the results of these experiments. The *linear / sibling d=4* performed surprisingly similar but a bit stabler than the Golden Standard. The other gloss based methods were always were in terms of MAP. The graph based concept extraction methods result in very poor performance for all combination methods.

[4]http://nlp.stanford.edu/software/tagger.shtml

| top-n | $\delta_{const}$ | $\delta_{lin}$ | $\delta_{exp}$ |
|---|---|---|---|
| 2 | 0.0190 | 0.0182 | 0.0156 |
| 4 | 0.0175 | 0.0168 | 0.0144 |
| 6 | 0.0165 | 0.0160 | 0.0138 |
| 8 | 0.0161 | 0.0156 | 0.0135 |

$\delta$ functions, averaged over all neighborhoods
(a)

| top-n | r | nh=2 | nh=5 | nh=8 |
|---|---|---|---|---|
| 2 | 0.0173 | 0.0114 | 0.0233 | 0.0223 |
| 4 | 0.0175 | 0.0119 | 0.0209 | 0.0196 |
| 6 | 0.0163 | 0.0113 | 0.0197 | 0.0186 |
| 8 | 0.0156 | 0.0110 | 0.0191 | 0.0181 |

$smooth$ with $\delta_{const}$ and different $nh$s against base score $r$
(b)

Table 3: Evaluation of smoothing parameters on MAP

Looking at the performance of the combination methods using the Golden Standard one can see that the best methods are $mult$ and $nm$. Surprisingly $mult$ is in average of all concept extraction methods the best. It should be noted that in average the variance of all combination methods is rather low. The results for the Golden Standard with Top-8 can be found in the lowest row of the table. Here the $sm$ method performs best, that leads to the conclusion the number of included concepts influences the performance of the combination method.

## 6 Conclusion & Future Work

In this paper we presented our approach to TRECVID's video retrieval search task and focused on two particular problems: 1) the conversion from queries in natural language format to a ranked list of concepts and 2) the combination of the returned concepts to improve the ranking of the shots. WordNet was chosen as mediator between the queries and concepts. Several algorithms utilizing WordNet's content (gloss-based approaches) and structure (graph-based approaches) were investigated. Given the ranked list of concepts, several combination and scoring methods were applied in order to gain insights into the optimal number of concepts to combine and the optimal scoring function.

While gloss-based concept extraction scored not considerably worse than the golden standard in the search task (with one gloss-based run regularly outperforming the golden standard), the graph-based approaches performed about $50\%$ worse. Using only the Top-2 concepts for scoring proved to be the most effective. For the Top-2 the our method $mult$ and $nm$ performed the best. The later uses the timely smoothed values of shots. $sm$, a method which only considers smoothed values, performed best for the Golden Standard using the 8 best concepts

There are several directions for future work. One important issue is how to deal with named entities - topic 149 (*Find shots of Condoleezza Rice*) is such an example. The graph-based concept extraction algorithms did not return a single concept, since the *Condoleezza Rice* does not appear in WordNet. It would therefore be beneficial to have access to an alternative knowledge source like Wikipedia or a newspaper corpus as a fall-back option. Furthermore, classifying the queries into several different types [Volkmer *et al.*, 2006] and creating query type dependent concept extraction and scoring algorithms can also help to alleviate the current problems.

The score combination algorithms presented so far do not adequately take into account the quality of the concept detectors. While some concepts are detectable with high precision (e.g. *Face* or *Sports*), others pose great difficulties (e.g. *Police Security*). This additional information can be exploited by weighting the concept scores according to the quality of the concept detectors.

## References


[Aly *et al.*, 2007] Robin Aly, Djoerd Hiemstra, and Roeland Ordelmann. Building detectors to support searches on combined semantic concepts. In *Proceedings of the SIGIR-MMIR: Multimedia Information Retrieval Workshop - New Challenges in Audio Visual Search* -, 2007. to be published.

[Banerjee and Pedersen, 2002] S. Banerjee and T. Pedersen. An adapted lesk algorithm for word sense disambiguation using wordnet. In *Proceedings of the Third International Conference on Intelligent Text Processing and Computational Linguistics*, pages 136–145, 2002.

[Budanitsky and Hirst, 2006] A. Budanitsky and G. Hirst. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47, 2006.

[Campbell *et al.*, 2006] Murray Campbell, Alexander Haubold, Shahram Ebadollahi, Dhiraj Joshi, Milind R. Naphade, Apostol Natsev, Joachim Seidl, John R. Smith, Katya Scheinberg, Jelena Tesic, and Lexing Xie. IBM research TRECVID-2006 video retrieval system. In *Proceedings of the 4th TRECVID Workshop*, 2006.

[Fellbaum, 1998] Christiane Fellbaum. *Wordnet: An Electronic Lexical Database*. The MIT Press, 1998.

[Hauptmann *et al.*, 2006] A.G. Hauptmann, R. Baron, M. Christel, R. Conescu, J. Gao, Q. Jin, W.-H. Lin, J.-Y. Pan, S. M. Stevens, R. Yan, J. Yang, and Y. Zhang. Cmu informedias TRECVID 2005 skirmishes. In *Proceedings of the 3rd TRECVID Workshop*, 2006.

[Hir, 1998] *WordNet: An electronic lexical database*, chapter Lexical chains as representations of context for the detection and correction of malapropisms, pages 305–332. MIT Press, 1998.

[Jiang and Conrath, 1997] J. Jiang and D. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings on International Conference on Research in Computational Linguistics*, pages 19–33, 1997.

[Koskela *et al.*, 2006] Markus Koskela, Peter Wilkins, Tomasz Adamek, Alan F. Smeaton, and Noel E. O'Connor. TRECVid 2006 experiments at dublin city university. In *Proceedings of the 4th TRECVID Workshop*, 2006.

[Lea, 1998] *WordNet: An electronic lexical database*, chapter Combining local context and WordNet similarity for word sense identification, pages 265–283. MIT Press, 1998.

[Lin, 1998] D. Lin. An information-theoretic definition of similarity. In *Proceedings of the International Conference on Machine Learning*, 1998.

| Top2 of | $add$ | $cbw$ | $mult$ | $n$ | $nm$ | $nw$ | $sa$ | $sm$ |
|---|---|---|---|---|---|---|---|---|
| Golden Standard | 0.0715 | 0.0615 | **0.0801** | 0.0588 | **0.0801** | 0.053 | **0.0621** | **0.0719** |
| uniform / sibling d=4 | 0.0525 | 0.0564 | 0.0548 | 0.0553 | 0.0548 | 0.055 | 0.051 | 0.0566 |
| linear / noSibling d=4 | 0.0515 | 0.0541 | 0.053 | 0.0567 | 0.053 | 0.0556 | 0.0481 | 0.0465 |
| linear / sibling d=4 | **0.0736** | **0.0741** | 0.0754 | **0.0713** | 0.0754 | **0.0710** | 0.0606 | 0.064 |
| uniform / noSibling d=0 | 0.0443 | 0.048 | 0.0413 | 0.0477 | 0.0387 | 0.048 | 0.0433 | 0.0341 |
| $rel_{HS}$ | 0.0342 | 0.0335 | 0.0316 | 0.0317 | 0.0316 | 0.0314 | 0.0334 | 0.0278 |
| $rel_{LC}$ | 0.0339 | 0.0332 | 0.0312 | 0.0315 | 0.0312 | 0.0312 | 0.0334 | 0.0278 |
| $rel_{WU}$ | 0.0339 | 0.0334 | 0.031 | 0.0316 | 0.031 | 0.0311 | 0.0329 | 0.0262 |
| Average: | 0.0494 | 0.0493 | ***0.0498*** | 0.0481 | 0.0495 | 0.047 | 0.0456 | 0.0443 |
| Golden Standard Top8 | 0.0600 | 0.0372 | 0.0593 | 0.0464 | 0.0593 | 0.0266 | 0.0408 | ***0.0681*** |

Table 4: MAP values for each Concept Extraction and Combination Method. **bold**: best for this combination method.


[Naphade and Smith, 2004] Milind R. Naphade and John R. Smith. On the detection of semantic concepts at trecvid. In *MULTIMEDIA '04: Proceedings of the 12th annual ACM international conference on Multimedia*, pages 660–667, New York, NY, USA, 2004. ACM Press.

[Naphade *et al.*, 2005] M.R. Naphade, L. Kennedy, J.R. Kender, S.-F. Chang, J.R. Smith, P. Over, and A. Hauptmann. A light scale concept ontology for multimedia understanding for trecvid 2005. Technical report, IBM T.J. Watson Research Center, 2005.

[Patwardhan and Pedersen, 2006] Siddharth Patwardhan and Ted Pedersen. Using wordnet-based context vectors to estimate the semantic relatedness of concepts. In *EACL*, 2006.

[Resnik, 1995] P. Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 448–453, 1995.

[Sebe, 2003] Nicu Sebe. The state of the art in image and video retrieval. In *Image and Video Retrieval*, volume Volume 2728/2003, pages 1–8. Springer Berlin / Heidelberg, 2003.

[Sigman and Cecchi, 2002] Mariano Sigman and Guillermo A. Cecchi. Global organization of the wordnet lexicon. *PNAS*, 99(3):1742–1747, 2002.

[Sjóberg *et al.*, 2006] Mats Sjóberg, Hannes Muurinen, Jorma Laaksonen, and Markus Koskela. PicSOM experiments in TRECVID 2006. In *Proceedings of the 4th TRECVID Workshop*, 2006.

[Smeaton *et al.*, 2006] Alan F. Smeaton, Paul Over, and Wessel Kraaij. Evaluation campaigns and trecvid. In *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, New York, NY, USA, 2006. ACM Press.

[Snoek *et al.*, 2006a] Cees G. M. Snoek, Jan C. van Gemert, Theo Gevers, Bouke Huurnink, Dennis C. Koelma, Michiel van Liempt, Ork de Rooij, Koen E. A. van de Sande, Frank J. Seinstra, Arnold W. M. Smeulders, Andrew H. C. Thean, Cor J. Veenman, and Marcel Worring. The MediaMill TRECVID 2006 semantic video search engine. In *Proceedings of the 4th TRECVID Workshop*, Gaithersburg, USA, November 2006.

[Snoek *et al.*, 2006b] Cees G. M. Snoek, Marcel Worring, Jan C. van Gemert, Jan-Mark Geusebroek, and Arnold W. M. Smeulders. The challenge problem for automated detection of 101 semantic concepts in multimedia. In *MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia*, pages 421–430, New York, NY, USA, 2006. ACM Press.

[Vapik, 1998] V. N. Vapik. *Learning Theory: Inference from Small Samples.* Wiley, 1998.

[Volkmer *et al.*, 2006] Timo Volkmer, S.M.M. Tahaghoghi, and James A. Thom. RMIT university video retrieval experiments at TRECVID 2006. In *Proceedings of the 4th TRECVID Workshop*, 2006.

[Wu and Palmer, 1994] Z. Wu and M. Palmer. Verb semantics and lexical selection. In *32nd Annual Meeting of the Association for Computational Linguistics*, pages 133–138, 1994.

[Yan, 2006] Rong Yan. *Probabilistic Models for Combining Diverse Knowledge Sources in Multimedia Retrieval*. PhD thesis, Canegie Mellon University, 2006.