Vidya: An Experiential Annotation System

Bageshree Shevade Hari Sundaram

Arts Media and Engineering Program
Arizona State University
AZ 85281
e-mail: {bageshree.shevade, hari.sundaram}@asu.edu

Abstract

In this paper, we present a novel annotation paradigm with an emphasis on two facets – (a) the end user experience and (b) semantic propagation. The annotation problem is important since media semantics play a key role in new multimedia applications. However, there is currently very little incentive for end users to annotate.

The annotation system, is interactive and experiential. We attempt to propagate semantics of the annotations, by using WordNet, a lexicographic arrangement of words, and low-level features extracted from the images. We introduce novel semantic dissimilarity measures, and propagation frameworks. The system provides insight to the user, by providing her with knowledge sources, that are constrained by the user and media context. The knowledge sources are presented using context-aware hyper-mediation.

Our Experimental results indicates that the systems performs well. We tested the new annotation experience using a pilot user study, the users agreed that the new framework was more useful that a traditional annotation interface. The semantic propagation results are good as well – we converge close to the semantics of the image by annotating a small number (~15%) of database images.

Categories and subject descriptors

H.3.1 [**Content Analysis and Indexing**]: *Abstracting methods, Dictionaries, Indexing methods, Linguistic processing,* H.5.5 [**Hypertext/Hypermedia**]: *Navigation.*

General terms

Algorithms, experimentation, human factors, languages, theory

Keywords

Experiential, annotation, semantic propagation, hyper-mediation, serendipity

1 Introduction

The goal of this is to create novel semi-automated, intelligent annotation algorithms that bridge manual methods for annotation and fully automatic techniques. This is a fundamental issue in multimedia because these algorithms need to rely on semantic level inferences in order to create a rich end-user experience [7,10,11,19].

Much work in the multimedia community has focused on automatically determining these semantics has focused either on traditional pattern recognition tools, or on content based retrieval techniques – these are very efficient, but the accuracy of such systems falls far below what is needed in commercial applications. At the other extreme is the traditional, fully manual annotation scheme – it is laborious and time consuming, but highly reliable. Clearly, manual annotation is the only reliable source of semantics in the context of multimedia data, and therefore crucial in construction of electronic experiences.

There has been prior work in semi-automatic image annotation using relevance feedback [5,6,8,17,18,22,23]. While there are rich mathematical models used, we believe that there are two shortcomings of the current work:

- **Experience:** None of the systems focus on the end-user experience. We believe that one of the barriers for users to annotate their media, is the absence of compelling and rich end user electronic experiences – there is very little return on the enormous time investment. Currently annotations only enhance the search capability and *not* the presentations. An incremental generation of an electronic experience will serve as feedback to the user, and prompting the user to enter richer annotations.
- **Semantics:** Current approaches to annotate images [5,8,18,20,22,23] essentially treat words as symbols regardless of their semantic relationships with other words, which is no different than any normal image feature. The lexical meaning of the keywords/annotations is not exploited.

There are also a number of tools and drag and drop interfaces available for annotating images, but all of these lack any kind of label propagation. For example, in the system described in [18], the user still has to drag and drop the labels for each and every image, even if there are repeated images of the same people at different events. This system is also cumbersome to use when confronted with media that have multiple semantic descriptors.

In this work we address issues relating to both semantics and the end user experience. We establish the semantic inter-relationships amongst the annotations by using WordNet [14], WordNet is a differential lexical organization of English developed by linguists. We additionally introduce a novel measure of semantic concept dissimilarity.

Annotation is a frustrating experience for most consumers. We have attempted to address this issues by mapping the annotation problem as one of an experiential system [10,19] – the key idea being that the user will gain insight about the media in relation to

ETP'03, November 7, 2003, Berkeley, California, USA.

the her context, thus providing a strong incentive for the user to annotate the media.

The annotation procedure is as follows. Our annotation system uses a combination of low-level as well as WordNet distances to propagate semantics. As the user begins to annotate the images, the system creates positive example (or negative examples) image sets for the associated WordNet meanings. These are then propagated to the entire database, using low-level features as well WordNet distances. The system then determines the image that is least likely to have been annotated correctly and presents the image to the user for relevance feedback. The media are displayed using a fisheye view (see Figure 1) and due to the change in the annotation as well in the user context due to interaction, the media changes its location to the center in real-time. The center represents the user context thus the movement signifies changes in the relationship between the semantics of the media and the user.

The system attempts to provide insight by presenting knowledge sources to the user. The underlying premise is that *serendipitous* activities lead to an increase in the user's *overall* knowledge. This is done using context-aware hyper-mediation [19]. Serendipity has also been explored in other contexts [4,15].

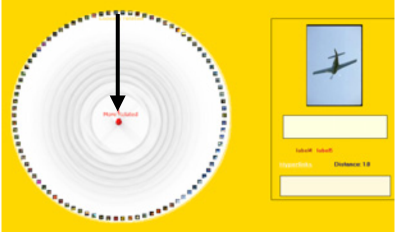


Figure 1: The proposed annotation interface. Initially, all the images are located at the periphery, indicating an infinite semantic distance from the user context. With user annotation, the system responds by offering knowledge sources related to the user and the annotation context. in real-time.

Hyper-mediation does two things – (a) It allows for a user-context constrained exploration of information related to media and (b) it introduces serendipity into the user's exploration of the media. In our approach, we use Google [2] to automatically generate hyper-links. This is done by taking into account the user profile, the semantics of the media and the semantic relationship between the media item and the user profile. We use the first hyper-link returned by Google. At the end of this procedure, we have a hyper-link for each non stop-word annotation of the media in the database. By monitoring the user activity using the system derived hyperlinks we can estimate changes to the user context. The experimental results on both the user experience and the semantic propagation indicate that the system is performing well.

The rest of this paper is organized as follows. In the next section, we present a brief overview of experiential systems. In section 3, we discuss the features used in our system. In section 4, we present details on the semantic propagation algorithm. Section 5 discusses the annotation interface and hyper-mediation while in section 6 we present the experimental results. And finally, we present our conclusions in section 7.

2 Experiential systems

An experience is commonly understood as follows:

> ***ex·pe·ri·ence:*** *the fact or state of having been affected by or gained knowledge through direct observation or participation [1].*

Clearly, this definition is narrow and does *not* adequately encompass the debates on the phenomenology of experience (e.g. as in [12]). The definition however, is useful in creating a computational basis for measuring changes in the knowledge state of the user.

Here are some specific examples of experiences – (a) a child learning to use Lego to build a house, (b) a visit to the beach, (c) the knowledge gained from learning to cook for the first time. Note that each experience involves a gain in knowledge, is multi-sensory, real-time and as well user-centric – it is dependent upon the specific knowledge that each user possess as well as the context of the experience. An experiential system is defined as follows:

> *Experiential systems are real-time, context-aware, user-centric and multi-modal. They cause a variation in the knowledge in the observer, due to direct interaction with the computer mediated environment [10,21].*

This work aims to develop computational models for annotation that are experiential. Experiential systems are *not* concerned with the problem of re-creating a natural experience. The experiential system, by re-mediating the original knowledge *augments* the knowledge (first-hand or indirect) that exists in the viewer, enabling the viewer to discover connections previously missed.

3 Features

In this section we shall discuss the low-level as well as the semantic features used in our annotation system.

3.1 Media Features

In this subsection we describe the low-level features used and the calculation of distance between two images based on these features. In this work, our feature only comprises color histograms as they are widely used and have proven to be fairly robust. In the future we plan on expanding the feature to include shape, texture among other features. We have used the HSV color space instead of the RGB color space as the HSV space is perceptually continuous. We begin by quantizing the RGB color space into 6 levels of red, 6 levels of green and 6 levels of blue as well as 16 grey levels. This RGB palette is converted to HSV using the standard conversion formula [9]. Hence the color histogram consists of 232 bins. The low-level color histogram distance between two images is given as follows:

$$d_{fv} = \mathbf{x}\Lambda\mathbf{x}^t, \qquad <1>$$

where **x** is the difference vector of the HSV color histogram of the two images being compared. Note that Λ is the color similarity matrix. The elements of Λ, $a(i,j)$ represent the similarity between colors i and j, and are given as $a(i,j) = 1 - d(i,j)$. $d(i,j)$ is the distance between colors i and j, and is given as follows:

$$d_c(i,j) = \frac{1}{\sqrt{5}}\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}, \qquad <2>$$

where, $x_i = s_i cos(2\pi h_i)$, $x_j = s_j cos(2\pi h_j)$, $y_i = s_i sin(2\pi h_i)$, $y_j = s_j sin(2\pi h_j)$, $z_i = v_i$, $z_j = v_j$ and (h_i, s_i, v_i) and (h_j, s_j, v_j) represent two HSV colors, where $h \in [0,1]$, $s \in [0,1]$, and $v \in [0,1]$.

3.2 Media Semantics

Semantics are incorporated and propagated in our framework through the use of WordNet ontology, which is an on-line lexical database [13]. WordNet organizes the English nouns, verbs and adjectives into synonym sets (synsets), which represent a lexical concept. For example, the English word "suit" can have multiple meanings such as a "pack of cards", "clothes", "a lawsuit or a petition", and can therefore belong to multiple synsets. Conversely, each synset has multiple words or word forms, which are synonyms of each other and which convey the same central meaning. These synsets are further organized into generalization (hypernyms) and specialization (hyponyms) or the is-a relationship, synonym and antonym, and holonym and meronym, or a part-whole relationship. The following Figure 2 illustrates the hierarchical relationship between synsets.

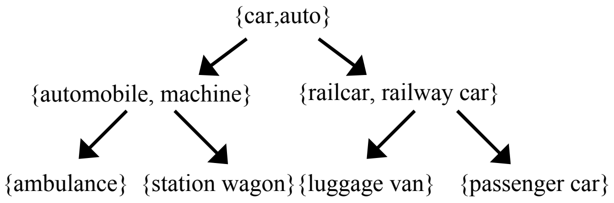


Figure 2: Hierarchical relationship between synsets

This hierarchical organization leads to the concept of implication/likelihood among words. For example, the word "Ambulance" implies the concept of "automobile" with certain likelihood and vice versa. Hence, if the user annotates an image with a keyword "Ambulance" and another keyword "automobile" is present in the database, then we can associate the word "automobile" to this image with certain likelihood. Similarly if the user is searching for images of "automobile" and the database contains images of "Ambulance", "Bus", "Taxi" etc. then these images can be shown in the search result because the WordNet hierarchy implies that an "Ambulance" is a kind of "automobile". We now present a mechanism to compute the implication distance between two synsets.

3.2.1 Semantic distance

In this section, we shall determine a procedure to compute the implication measure between any two synsets in the wordnet hierarchy. The measure is *not* a metric, since the wordnet has a specific generalization/specialization relationship associated with it. This causes the symmetry requirement of a metric not to hold. We shall compute dissimilarity using an intermediate notion of implication. The distance between two wordnet synsets will determined using the implication measure.

A concept α (e.g. fruit) has two entities associated with it – the parent concept, and the children concepts. Each entity implies the concept with a different weight – ω_1 (the parent) and ω_2 (all the children). Hence the *implication* that the concept α is true given that another concept β is true is computed as follows:

$$\begin{aligned} I(\alpha \mid \beta = T) &= \omega_1 I(parent \mid \beta = T) + \frac{\omega_2}{k}\sum_{i=1}^{k} I(c_i \mid \beta = T), \\ I(\beta \mid \beta = T) &= 1, \\ \omega_1 + \omega_2 &= 1, \end{aligned} \quad \text{<3>}$$

where I is the implication strength, and k is the number of children of the concept α, c_i is the i^{th} child of the concept, and where ω_1 and ω_2 are the weights attached to the implications of the parents and the children respectively. Note the following:

1. Implications of the parent and the children nodes are evaluated independently of the concept α, to prevent cycles.
2. Implications of the root node, or leaf node with respect to concept β can only be equal 1 in the case of identity:

$$I(\alpha \mid \beta = T) = \begin{cases} 1 & \alpha = \beta \\ 0 & \alpha \neq \beta \end{cases} \quad \text{<4>}$$

   where α is a either a root node or a leaf node in WordNet.

3. The weight ω_1 is computed to be inversely proportional to the number of children of the *parent* concept, α. i.e.

$$\omega_1 \propto 1/m \quad \text{<5>}$$

   where m is the number of children of the parent.

4. The distance between the two concepts is now easily determined as follows:

$$\begin{aligned} d_U &= 1 - I(\alpha \mid \beta = T) \\ d(\alpha \mid \beta = T) &= d_U / \sqrt{f_\alpha \bullet f_\beta} \end{aligned} \quad \text{<6>}$$

where d_U is the un-weighted distance between the two concepts. f_α and f_β, represent normalized knowledge priors for concepts α and β. The priors are used to re-weight the distance. These could be set by the user, as part of her context model. If these values are unknown we just use $f_\alpha = f_\beta = 1.0$. Note that these values could also be determined using the normalized frequency of occurrence of the concept.

Again, the measure is anti-symmetric, as expected. This recursive formulation is (ref. equation <3>) is computationally expensive, so in practice we approximate the implication value using a pruned tree [19].

4 Propagation of Semantics

In this section we shall discuss the algorithm for the propagation of semantics. The problem concerns WordNet synsets as well as a set of images. Each synset is associated with positive and negative example images. These synsets and their positive and negative example images arise due to user feedback. Let us consider the following example. Let us assume that the database consists of two synsets "flower" and "sunrise", with at least one user annotated image associated with each. This is shown in Figure 3.

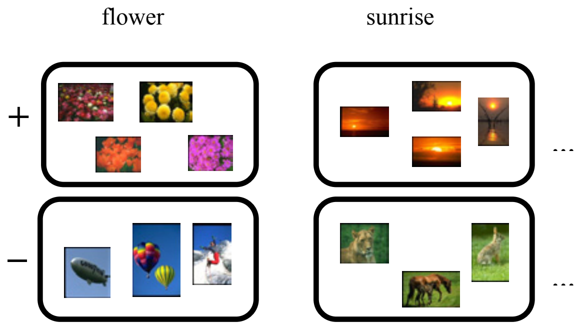


Figure 3: Positive and Negative example images associated with synsets

Let us assume that the user wishes to annotate an image. She enters the annotation for the image and fixes the sense of the word to be the synset "rose".

We now want to propagate this newly added synset to all the other images in the database. Hence we need to calculate the likelihood that an existing image is associated with this new synset. This likelihood is computed by calculating the low-level feature distance of the query image (the image to which we want to propagate the synset) from each synset as well as by calculating the WordNet distance.

In our example, we compute the low-level distances by comparing the color histogram of every other image in the database, with the positive and negative example images of synset "rose". We then compute WordNet distances between the synsets "rose" and "flower", and "rose" and "sunrise", and propagate the synset "flower" and "sunrise" to the image of "rose". After this propagation the system picks the image that is least likely to be associated with the annotations that accurately reflect the semantics of the image. The user provides relevance feedback for this image. The user can add a new synset; confirm or delete an associated synset. The system then updates the positive and negative example images associated with each synset and again propagates the changes throughout the database. Thus with each iteration, the association between the annotations and the images gets more refined and reflects the image semantics. The following Figure 4 is a flow chart of the algorithm, which is explained in detail in the next section.

Thus, if we follow the cycle depicted by the above diagram, then after k iterations, we would have M synsets. Note that in general $k \neq$ M as during each iteration, the user can add; delete or confirm one or more synsets associated with an image. At the end of the cycle, each synset will have positive and negative examples associated with it.

4.1 Algorithm Details

In this section, we present the details of our algorithm on semantic propagation. Let us assume that the user wishes to annotate an image a and that there are N images in the database.

When the user enters annotations for image a, she is asked to fix the sense (i.e. the semantics) of the word that she is using for annotation using WordNet. Fixing the sense of the word exploits the hierarchical relationship among synsets in WordNet. The current image a is considered as a positive example image for this fixed synset.

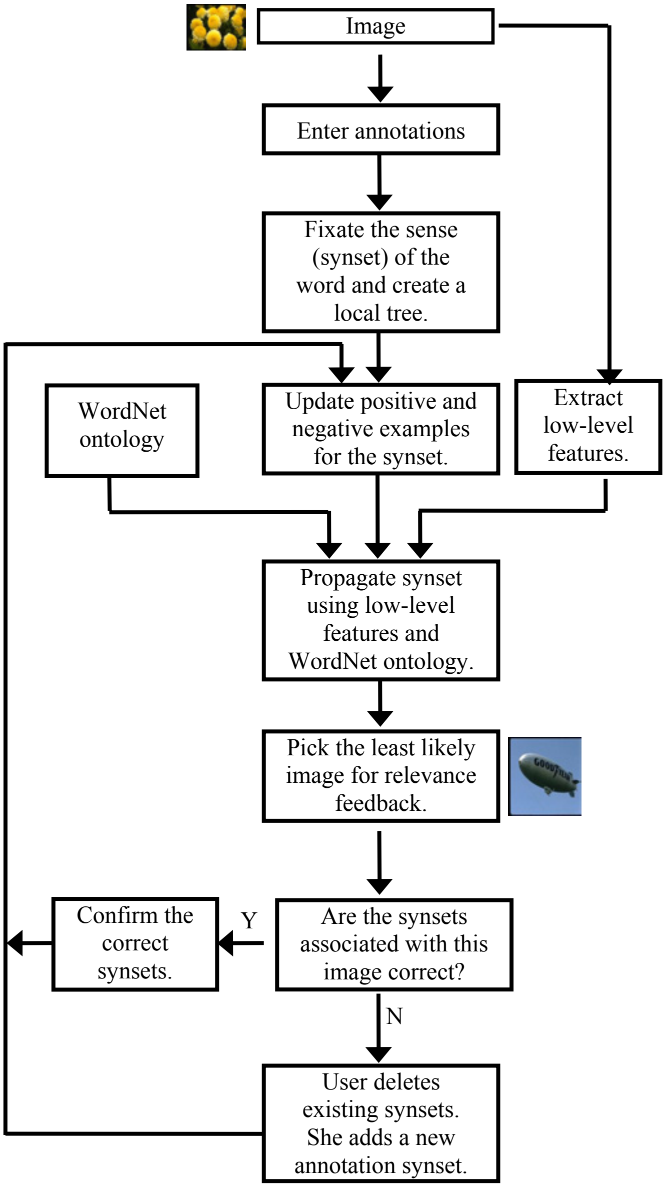


Image
Enter annotations
Fixate the sense (synset) of the word and create a local tree.
WordNet ontology
Update positive and negative examples for the synset.
Extract low-level features.
Propagate synset using low-level features and WordNet ontology.
Pick the least likely image for relevance feedback.
Confirm the correct synsets.
Y
Are the synsets associated with this image correct?
N
User deletes existing synsets. She adds a new annotation synset.

Figure 4: Flow Control

4.1.1 Local trees

For each fixed synset k, we need to define a local tree T_k. A local tree is a subset of the WordNet ontology. It is a hierarchy of nodes, with synset k being the node at the center of the hierarchy. Synset k is also called the root node of the tree. Nodes above the center are the generalizations of synset k and nodes below it are its specializations. Formally, a local tree T_k is defined as follows:

$$T_k = \{s \mid d(s,k) \leq m\}, \qquad <7>$$

where $d(s,k)$ is the hop distance between the node representing synset s and synset k, and m is the number of specialization and generalization levels to be considered. In our case, we set m = 2, based on a trade off between computational complexity and accuracy. Thus a local tree efficiently partitions the semantic space of WordNet. It helps to quickly identify if two synsets are semantically far apart. For example, if two synsets are not in each other's local tree, we infer that the WordNet distance between these two synsets is infinity.

4.1.2 Computing semantic propagation likelihoods

The new synset k, entered by the user is then propagated to other images based on low-level features using color histograms and WordNet using local trees. This is done as follows: we determine $L_f(k|i)$, that is the low-level feature likelihood that image i belongs to synset k.

$$L_f(k \mid i) = L_f(k^+ \mid i) - L_f(k^- \mid i), \qquad <8>$$

where, k^+ denotes the set of all the positive example images directly associated with synset k, as well as the positive example images associated with the synsets present in the local tree of k. k^- denotes the set of all the negative example images of synset k, as well as all the positive example images of the synsets, not present in the local tree of k. $L_f(k^+|i)$ denotes the low-level likelihood that the image i belongs to the set k^+. $L_f(k^-|i)$ denotes the low-level likelihood that the image i belongs to the set k^-. Equation <8> is intuitive because the likelihood that an image is associated with a synset ought to be the difference between the likelihood that the image is close to the positive examples of the synset and the likelihood that the image is close to the negative examples of the synset.

$L_f(k^+|i)$ is defined as follows:

$$L_f(k^+ \mid i) = \sum_{j=1}^{M} \exp(-\beta d_{ij}) I(s_j, k), \quad s_j \in T_k, \qquad <9>$$

where M is the total number of synsets in T_k (local tree of synset k), whose positive examples are being considered. d_{ij} is the average low-level feature distance between image i and the positive examples of synset s_j. β is a constant and $I(s_j,k)$ is the WordNet implication between synsets s_j and k. (ref. section 3.2.1). Equation <9> is intuitive since the likelihood that the image is close to the set k^+ ought to be proportional to the sum of the likelihoods that the image is close to the positive examples of synsets present in the local tree of k. These likelihoods should be modified by the relationship between the synsets present in the local tree and the root node of the local tree. This must be, since we are trying to compute likelihood with respect to the root node.

$L_f(k^-|i)$, denotes the low-level likelihood that the image i belongs to set k^- and is given as follows:

$$L_f(k^- \mid i) = \omega_1 \exp(-\beta d_{ik}) + \frac{\omega_2}{M} \sum_{j=1}^{M} L_f(s_j^{\,+} \mid i), \quad s_j \notin T_k, \qquad <10>$$

where β is a constant, d_{ik} is the average low-level feature distance between image i and the negative examples of synset k. ω_1 and ω_2 are weights where $\omega_1+\omega_2=1$ and M is the total number of synsets not in the local tree of synset k. Equation <10> is intuitive since the likelihood that the image is close to the set k^- ought to increase if the image is either close to the negative examples of synset k or close to the positive examples of synsets not in the local tree of k.

We now show how to compute the WordNet likelihood. This is done by calculating the likelihood of other synsets already present in the database (but not manually associated with the image) to image a. These other synsets are present in the database, as the user has entered them as manual annotations for some other images in the database. This likelihood is the WordNet likelihood and is given as follows:

$$W_I(s_1, s_2, \ldots s_M \mid k) = \frac{1}{M} \sum_{j=1}^{M} I(k, s_j), \qquad <11>$$
$$L_w(k \mid i) = W_I(s_1, s_2, \ldots s_M \mid k)$$

where $s_1,s_2 \ldots s_M$ are the synsets which are manually associated with the image a by the user. $I(k,s_j)$ is the WordNet implication between synset k which we want to propagate and manual synset s_j and M is the total number of synsets manually associated with image a.

4.1.3 Feedback

The system now presents an image to the user for relevance feedback. This is an image that is least likely to be associated with the annotations that accurately reflect the semantics of the image. The final likelihood, for picking the least likely image, for an image i is computed as follows:

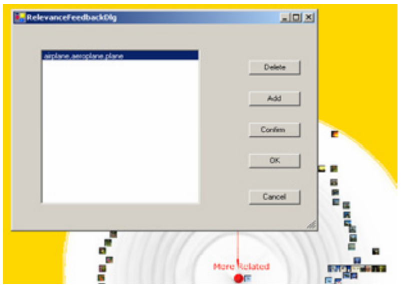
$$l_i = \frac{1}{M} \sum_{j=1}^{M} \alpha L_f(s_j \mid i) + \beta L_w(s_j \mid i), \qquad <12>$$

where M is the number of synsets which were automatically propagated to image i. $L_f(s_j|i)$ is the low-level likelihood of image i with respect to synset s_j and $L_w(s_j|i)$ is the WordNet likelihood of image i with respect to synset s_j. α and β are constants where α+β=1. The least likely image j* is picked as follows:

$$j^* = \arg\min_j l_j, \qquad <13>$$

where j varies over the total number of images in the database, and l_j is the final likelihood of image j.

During relevance feedback, the user can either delete an associated synset, or confirm an automatically propagated synset or add another synset as shown in Figure 5.

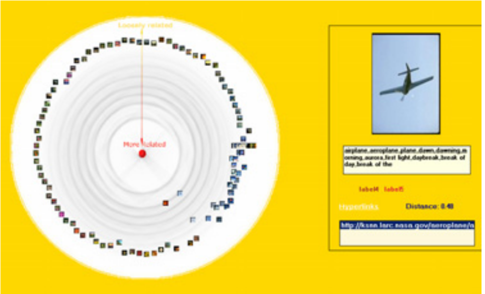


airplane,aeroplane,plane

Figure 5: User Interface for relevance feedback

If the user deletes an associated synset, then we mark the image as a negative example of the synset. We then recompute the distance of every other image in the database, with respect to the newly updated positive and negative example sets of that synset. On the other hand, if the user confirms an automatically associated synset, then we consider this image as a positive example of the synset, and follow the same procedure of recomputing the distance of every other image, as explained earlier. Same holds true if the user adds a synset to the image. Thus with each iteration of relevance feedback the association between images and synsets gets refined, and becomes more accurate with respect to the image semantics.

5 The Annotation Interface



airplane,aeroplane,plane,dawn,dawning,morning,aurora,first light,daybreak,break of day,break of the

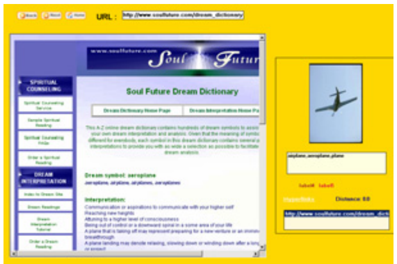
Distance: 0.48

http://ksnn.larc.nasa.gov/aeroplane/a

Figure 6: User Interface for creating hyper-mediated experiences. The distance of the image from the center represents the relationship between the semantics of the image and the user profile. The images away from the periphery to the center based user annotation and changes in the user context.

We now present our framework to create an experiential annotation framework. This assumes the existence of a user profile. From this user profile, dominant concepts are extracted and stored in the form of a concept graph [19]. Every non-stop word that is a noun in the user profile becomes an elementary concept. If a group of elementary concepts belong to one lexical concept, then that lexical concept is said to be the dominant concept. A graph consisting of these dominant concepts as nodes, and the relationship between these concepts as links, is said to be a concept graph.

The concept graph is used to create the hyper-links in the annotations. It determines what words in the annotation are hyper-linked and where they are hyper-linked, depending on the concepts in the user profile and the concepts present in the annotations. For every image, the top three annotations whose likelihood is above a certain threshold are considered for hyperlinking. For hyperlinking, if the annotation is a proper noun, then it is compared with the elementary concepts in the user's profile, if it matches, then the search query for generating the hyperlink becomes the proper noun and the dominant concept of the proper noun. However, if it does not match any elementary concept in the user profile, then the search query becomes only the proper noun. Similarly, if the annotation is a common noun, then it is compared with all the elementary concepts in the user profile. If it matches any one of them, then the search query becomes that word and the dominant concept of the elementary concept and if it does not match, then we calculate the WordNet distance of that annotation from every dominant concept and pick a random elementary concept under the closest dominant concept and use it in the search query along with the annotation.



http://www.soulfuture.com/dream_dictionary

Soul Future Dream Dictionary

This A-Z online dream dictionary contains hundreds of dream symbols to assist your own dream interpretation and analysis. Given that the meaning of symbo different for everybody, each symbol in this dream dictionary contains several p interpretations to provide you with as wide a selection as possible to facilitate dream analysis.

Dream symbol: aeroplane

aeroplane, airplane, airplanes, aeroplanes

Interpretation:

Communication or aspirations to communicate with your higher self
Reaching new heights
Attuning to a higher level of consciousness
Being out of control or a downward spiral in a some area of your life
A plane that is taking off may represent preparing for a new venture or an immin breakthrough
A plane landing may denote relaxing, slowing down or winding down after a lon or project

airplane,aeroplane,plane

Distance: 0.0

http://www.soulfuture.com/dream_dicti

Figure 7: Navigating to the hyper-links.

The user interface that enables the creation of hyper-mediated experiences is shown in Figure 6. The center of the circle denotes the user profile. Images closer to the user profile are closer to the center and vice versa. This interface was co-developed as a part of [16]. It allows the user to walk through all the media objects present in the database. It also helps searching for an object that is close to the user profile and is real-time.

Initially all the un-annotated images are situated at the periphery of the circle, which signifies a distance of infinity from the user profile. Now as the user selects an image and annotates it, these annotations propagate throughout the entire database of images. As a result, distance of each image with respect to the user profile changes, depending on the concepts present in the annotations and the user profile, and the images move closer or away from the center of the circle, signifying how these images are related to the user profile.

For each image, this distance is the semantic distance between the image and the user profile. It is calculated by computing the likelihoods of each synset associated with the image modified by the WordNet likelihood of the synset and the user profile, and is given as follows:

$$D = 1 - \frac{1}{M}\sum_{j=1}^{M} L(s_j \mid i) I_p(s_j, p),$$

$$L(s_j \mid i) = L_f(s_j \mid i) + L_w(s_j \mid i), \qquad <14>$$

$$I_p(s_j, p) = \sum_{i=1}^{r} I(s_j, p_i)$$

where M is the number of distinct synsets in the database and *p* is the user profile. $I_p(s_j,p)$ is the maximum WordNet implication of the synset s_j with respect to the concepts in the user profile and $L(s_j|i)$ is both, the low-level likelihood and the WordNet likelihood of image *i* with respect to the synset s_j and *r* is the total number of concepts present in the user profile. When the user selects an image, the right panel shows the details viz. the annotations and the hyper-links as well as the distance of the image from the user profile. The user can click on the hyper-links and wander off to the internet. This is shown in Figure 6. In our example, the user profile mentioned flying of aircrafts as the user's dream; hence the hyper-link in Figure 7 talks about aircrafts as dream symbols.

The interface also allows the user to provide relevance feedback for the annotations as shown in Figure 5. It makes the process of annotation much more interesting and encouraging to the user, which results in a richly annotated database of images, which in turn produces more accurate keyword-based image retrieval search results.

6 Experiments

The experiments were conducted on a set of 242 images. The ground truth for these images was created manually, by fixing the sense of the annotation. The entire prototype was developed in Visual J# in Microsoft Windows platform.

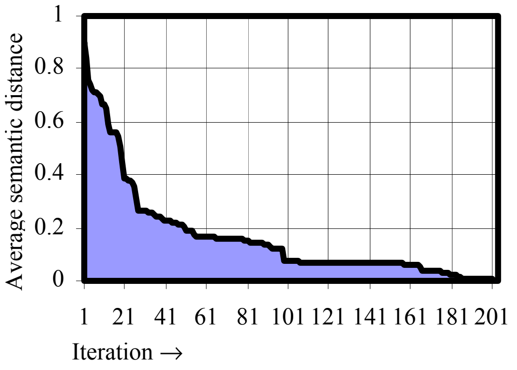


Figure 8: This figures shows the average semantic distance of the entire database, from the ground truth annotations. In each iteration, only *one* image is annotated. The figure shows that by annotation just 41/242 (~16%) images, we can decrease the semantic distance to 0.2

In order to test the system, we created an automatic test script, which initially picked a random image. We exposed the ground truth of this random image and annotated it with the ground truth synset. This is equivalent to the user picking an image and annotating it. After performing all the required propagations, the system picked an image that was least likely to be associated accurately, with the semantics of its annotations, and uncovered its ground truth. If the automatic annotation synsets of this image coincide with the ground truth, then the system confirms the automatic annotations and updates the positive example images for that synset. However, if the automatic synsets and the ground truth synsets don't match, i.e. they are not present in each other's local tree, then the system considers the image as a negative example of the automatically propagated synsets, and a positive example of the ground truth synset. This is equivalent to the user giving relevance feedback for the least likely image, where he deletes the automatically propagated synsets, and adds the ground truth synsets to the image..

In our current prototype, the user only gives feedback on *one* image. This was done to mimic an ordinary end-user. This image is the one which is least likely to be associated with the annotations that accurately reflect the semantics of the image. After the automatic relevance feedback, the system, again performs the required propagations, picks the least likely image and executes the entire cycle explained above. We executed the cycle mentioned above for 200 iterations

6.1 Evaluation

We now describe our evaluation mechanism. During each iteration, the WordNet distance between the propagated image label, and ground truth annotation is calculated for all images. The average of these distances is then used to compute the correctness of the propagation algorithm. The average WordNet distance at the end of each iteration, is calculated as follows:

$$\bar{D}_w = \sum_{i=1}^{N} \min_{j,k} d_w(a_{i,j}, g_{i,k}) \ , \qquad <15>$$

for all images, where $a_{i,j}$ is the automatic annotation associated with an image i, $g_{i,k}$ is the ground truth annotation for image i, N is the total number of images in the database and $d_w(a_{i,j},g_{i,k})$ is the WordNet distance between the two synsets $a_{i,j}$ and $g_{i,k}$. Figure shows the graph of average WordNet distance against number of iterations for one such random image.

As the graph (ref. Figure) suggests, the semantic distance between the annotations and the images decreases with the increase in the number of iterations of relevance feedback. The graph represents the average of 15 experiments, each carried out with a different random image. We did not use precision and recall, the conventional metrics, since they do not reveal the complexities of the semantic propagation – for example, an image of an "apple", mislabeled as "fruit" is *not* a large error as is intuitive. This is intuition is based on the semantic relationships as established by our culture and is captured well by WordNet.

Our results compare well with related work [5,23]. For example, in [23] the authors show that the algorithm converges to 90% accuracy within four iterations. However, in *each* iteration, the system evaluates a 100 images for relevance feedback. In our system, we achieve a WordNet distance of 0.2, using only 41 iterations. In [5] the authors report achieving a 50% accuracy by annotating only 20% of the images; this compares well with our result. However, since the authors use only class membership and not WordNet to present their results, we believe that their results will improve with the use of WordNet.

6.2 User study

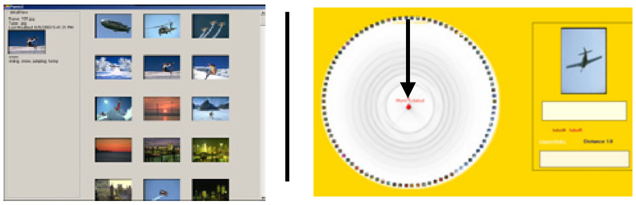


Figure 9: The user study was compared two interfaces – a "traditional" windows explorer based annotation style against the our proposed system.

We also conducted a user study to compare the annotation interface described in this work, with the default annotation interface, which involves no label propagation and no relevance feedback. Both the interfaces are shown in Figure 9.

Table 1: The user study results

Question	Score (1-7)
Is the system annotation interface more encouraging to annotate than the traditional UI ?	**6.6** / 7.0
Hyper-link quality: Do they reflect your interests?	**5.0** / 7.0
Does the movement of images hinder the process of annotation [1-7]?	**2.4** / 7.0

We asked users to create a user profile, which was dependent on the context of the database. This was done to ensure the creation of hyper-links. Each user was explained the usage and motive, behind both the interfaces. Every user was given a total time of 30 minutes, to annotate the database, using both the interfaces. The user was free to choose the amount of time spent on each interface. At the end of 30 minutes, each user was asked three questions. The results of the user study are shown in Table 1and are averaged over five users.

As the results indicate, most of the users found the annotation interface, described in this work, much more intuitive and encouraging to annotate. Since it involved label propagation, users did not have to select each and every image, in order to annotate. This is unlike the default annotation interface, which seems tedious and time-consuming. The default interface, also did not allow fixing the sense of the word for the case when a word had multiple senses. In addition, the hyper-links that were generated were specific to each user and they did reflect the user's interest 70% of the time. While the results are encouraging, we need to conduct more extensive experiments over larger datasets coupled with more user studies.

7 Conclusions

In this paper, we have presented a novel strategy for annotating images and propagating these annotations across a large database. The propagation is done using the following – (a) low-level features (b) WordNet ontology and (c) relevance feedback. The framework also incorporated the idea of providing a hyper-mediated experience to the user, which makes the process of annotation much more exciting and interesting and is also real-time. We explained the algorithm for label propagation in detail, and also described the user interface which provided the hyper-mediated experience.

We then conducted an experiment to show how the semantics propagate across the database. The results indicate that as the number of relevance feedback cycles increases, the association between the images and the annotations, as far as semantics are concerned, becomes more and more refined and accurate. We also conducted a user study on the interface to determine its effectiveness in aiding users to annotate images. The results of the user study show that this interface is much more encouraging to annotate and it does create hyper-mediated experiences for the user as compared to the default annotation interface.

The algorithm could be improved in several ways. We plan in incorporating more features into the system. We are also looking at incorporating personalized ontologies into the system, since WordNet may be modied per user (i.e. the distances between concepts will differ on a per user basis.) Finally, we are also planning to incorporate facts using OpenCyc [3] to further enhance label propagation.

8 References

[1] *Merriam Webster Dictionary* http://www.m-w.com.

[2] *Google* http://www.google.com.

[3] *OpenCyc* http://www.opencyc.org.

[4] V. BUSH (1945). *As We May Think*. The Atlantic Monthly. **176:** 101-108, http://www.theatlantic.com/unbound/flashbks/computer/bushf.htm.

[5] E. CHANG, K. GOH, G. SYCHAY, et al. (2003). *CBSA: content-based soft annotation for multimodal image retrieval using Bayes point machines*. IEEE Transactions on Circuits and Systems for Video Technology **13**(1): 26-38.

[6] I. J. COX, T. V. PAPATHOMAS, J. GHOSN, et al. (1997). *Hidden Annotation in Content Based Image Retrieval*, Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries, 76-81,

[7] C. DORAI, A. MAUTHE, F. NACK, et al. (2002). *Media Semantics: Who Needs It and Why?*, Proceedings of the ACM Multimedia 2002, Juan-les-Pins, France, pp. 580-583, Dec. 2002.

[8] X. HE, W.-Y. MA, O. KING, et al. (2002). *Learning and inferring a semantic space from user's relevance feedback for image retrieval*, Proc. of the 10th international conference on Multimedia, Juan Les-Pins, France, 343-346, Dec. 2002.

[9] A. K. JAIN (1989). Fundamentals of digital image processing. Englewood Cliffs, NJ, Prentice Hall.

[10] R. JAIN (2003). *Experiential Computing*. Communications of the ACM **46**(7): 48-55.

[11] R. JAIN (2003). *Folk Computing*. Communications of the ACM **46**(4): 27-29.

[12] I. KANT, P. GUYER and A. W. WOOD (1998). Critique of pure reason. Cambridge ; New York, Cambridge University Press.

[13] G. A. MILLER, R.BECKWITH and C.FELLBAUM *Introduction to WordNet : An on-Line Lexical Database*. International Journal of Lexicography **3**(4): 235-244.

[14] G. A. MILLER, R.BECKWITH and C.FELLBAUM (1993). *Introduction to WordNet : An on-Line Lexical Database*. International Journal of Lexicography **3**(4): 235-244.

[15] T. H. NELSON (1987). Computer lib/dream machines. Redmond, Wash., Tempus Books of Microsoft Press.

[16] J. RAMAMOORTHY, S. WONG and H. SUNDARAM (2003). *Dynamic Adaptive Visualizations*. Arts Media and Engineering Center, ASU, AME-TR-2003-03, 2003.

[17] Y. RUI and T. HUANG (1999). *A Novel Relevance Feedback Technique in Image Retrieval.*, ACM Multimedia 1999,

[18] B. SHNEIDERMAN and H. KANG (2000). *Direct Annotation: A Drag-and-Drop Strategy for Labeling Photos*, In: Proc. International Conference Information Visualization (IV2000). London, England,

[19] H. SRIDHARAN, H. SUNDARAM and T. RIKAKIS (2003). *Context, memory and Hyper-mediation in Experiential Systems*. Arts Media and Engineering Center, ASU, AME-TR-2003-02, 2003.

[20] Y. SUN, H. ZHANG, L. ZHANG, et al. (2002). *A System for Home Photo Management and Processing*, Proceedings of the 10th ACM international conference on Multimedia, Juan Les-Pins, France, pp. 81-82, Dec. 2002.

[21] H. SUNDARAM and T. RIKAKIS (2003). *An Introduction to Experiential Systems*. Arts Media and Engineering Center, ASU, AME-2003-01, 2003.

[22] L. WENYIN, S. DUMAIS, Y. SEN, et al. (2001). *Semi-Automatic Image Annotation*, Proc. Human-Computer Interaction--Interact 01, pp.326-333,

[23] L. YE, C. HU, X. ZHU, et al. (2000). *A Unified Framework for Semantics and Feature Based Relevance Feedback in Image Retrieval Systems.*, In: Proc. ACM MM2000, 31-38,