Cross modal disambiguation

Kobus Barnard[1], Keiji Yanai[2], Matthew Johnson[3], and Prasad Gabbur[4]

[1] Department of Computer Science, University of Arizona,
kobus@cs.arizona.edu
[2] Department of Computer Science, The University of Electro-Communications,
1-5-1 Chofugaoka, Chofu-shi, Tokyo, 182-8585 JAPAN
yanai@cs.uec.ac.jp
[3] Department of Engineering, University of Cambridge
mj293@cam.ac.uk
[4] Electrical and Computer Engineering,
University of Arizona,
pgsangam@ece.arizona.edu

Abstract. We consider strategies for reducing ambiguity in multi-modal data, particularly in the domain of images and text. Large data sets containing images with associated text (and vice versa) are readily available, and recent work has exploited such data to learn models for linking visual elements to semantics. This requires addressing a correspondence ambiguity because it is generally not known which parts of the images connect with which language elements. In this paper we first discuss using language processing to reduce correspondence ambiguity in loosely labeled image data. We then consider a similar problem of using visual correlates to reduce ambiguity in text with associated images. Only rudimentary image understanding is needed for this task because the image only needs to help differentiate between a limited set of choices, namely the senses of a particular word.

1 Introduction

Recent work suggests that the semantics of images and associated text can be better learned from data if they are considered together. For example, to build a system for searching and browsing large data sets, one should take advantage of available textual information. However, text alone cannot capture all that is of interest in an image. Furthermore, images with detailed text descriptions are rare. Thus there has been recent interest in integrating available text with visual information. This includes providing methods for searching and browsing which use both image features and text [21, 22], and learning links between visual representations and words from loosely labeled training data [13, 25, 10, 20]. In this paradigm, the models learned can be used to add labels to new images (auto-annotate), or even image regions (region-labeling). Alternatively, the links can be implicit, and simply help queries based on visual descriptors to return more semantically meaningful results [13, 46].

The underlying key idea in these methods is the observation that images with associated text have substantive supervisory information that can be exploited. The main confound is ambiguity. For example, in an image labeled with the words "tiger", "water", and "grass", it is not known which parts of the image correspond to which of these words. The work cited above addresses this correspondence ambiguity by building models for the various visual concepts that are consistent over a number of images. In our example, the single image does not have sufficient information to determine which words go with which features. However, additional images with, for example, tigers without water, and water without tigers, the ambiguity can be reduced. The process of reducing the ambiguity by using large training sets is analogous to statistical machine translation (see figure 3 in chapter XXX).

Now consider a program for automatically labeling our example image based on a learned model. Labeling images is clearly a difficult task. However, it becomes easier if we assume that the labels must come from the associated words. In our example, this means that instead of choosing among potentially hundreds, or thousands of words, we only need to choose between three of them.

This constrained labeling of the training data is implicit in some of the learning approaches mentioned above. However, we find it useful to consider it more explicitly. Doing so emphasizes that there are two parts of the problem. First, we wish to migrate semi-supervised data towards supervised data. This is important if we are to use large, loosely labeled data sets in a more supervisory fashion. Second, we need to develop algorithms and models that are targeted for inference on new data. As mentioned above, current approaches deal with the dependence between these two problems by iteratively solving one and then the other. However, as the required models and inference become more complex, it may be beneficial to consider the tasks separately. For example, a simple model may be able to give a reasonable approximate labeling of training data. This labeling can then be used to develop inference approaches which might be difficult to integrate into the initial labeling method. Further, augmenting strategies, such as integrating supervisory data and language modeling, can be simplified if we explicitly reduce correspondence ambiguity in the training data first, and then build models for inference.

In this paper, we will suggest how language models can be used to reduce correspondence ambiguity. In the work reported so far, language models have been limited to a "bag of words" model. Further, the words are generally assumed to be nouns. However, different parts of speech such as nouns, adjectives, and prepositions relate to visual attributes differently. Further, since modern parts of speech tagging [16, 17] is relatively effective, there is opportunity to better exploit associated text through language tools. For example, certain (visual) adjectives embody specific image region features, and this is assumed to be consistent over multiple objects. If this relationship is known, it can help resolve the correspondence between words and image regions. Thus one can simultaneously learn the meaning of words such as "red", and use natural language analysis to exploit the occurrence of the modifier "red" to help learn the meaning of "ball"

from an image annotated with "red ball". Similarly, if we assume that certain simple prepositions reflect spatial relations, we should be able to simultaneously learn the meaning of those prepositions, and exploit that meaning to help learn the visual representation of nouns being spatially related.

Reciprocally, images can also help disambiguate language meaning. In particular, words in natural language are ambiguous because they have multiple meanings (senses). For example, the word "bank" has a number of meanings including "financial institution" and that suggested by "river bank". Intuitively, an image could help determine the senses in a sentence like: "He ate his lunch by the bank". All that is required is that we have an image that is more correlated with the correct sense. The image need not even contain a bank, nor do we need to identify banks; the image features only need to correlate better with the correct sense as compared with the incorrect sense.

It is important that a complete understanding of the image is not required, as this would make the approach impractical given the current state of automated and image understanding. Notice that the disambiguation task is made much easier because we only need to select among a limited number of choices; namely the senses of the word being considered. Again, the disambiguation task is simpler than a complete understanding, but reducing the ambiguity can help move towards an understanding.

In what follows we first review recently developed approaches for dealing with multi-modal data with correspondence ambiguity. We then consider two instances of cross modality disambiguation in further detail. Here we discuss how adjectives can reduce correspondence ambiguity in images with associated text. We also propose a method to prune adjectives that are not visual, relative to our features. Finally, we outline a method for using images to disambiguate words in natural language.

2 Matching Words and Pictures

A number of methods have been recently developed for predicting words from image data, based on a large training data of images with associated text. Critically, the correspondence between particular words and particular visual elements is not required, as large quantities of such data is not readily available and expensive to obtain. Current approaches include:

- Simultaneously learning a model and reducing ambiguity, with latent entities (concepts) competing with each other for image elements and words [13, 25, 10, 20]. This competition means that an image element that is more likely to be associated with one word (e.g. "tiger") is less likely to be associated with another one (e.g. "water"). Included here are translation approaches which constructs a model for words conditioned on image elements.
- Cross-media relevance models which predict words for entire images (auto-annotation) based on a statistical match of the image with components in the training data [35, 28].

- Multiple instance learning which builds a separate classifier for the presence or absence of each word in the vocabulary in the face of multiple possibilities of which image element is relevant [38, 39, 4, 52, 53, 5, 6]. While not explicitly developed to do so, these methods support region labeling, and have recently been evaluated on this task [12].
- Object category recognition efforts [15, 29, 27, 48], which are focused on identifying the existence of an object category, are related to the task of predicting words for images, and could be evaluated in the same way. Here the data is typically of an instance of an object category, with non-trivial clutter.

Here we review one method from the first approach which we build on below. Specifically we will consider the *dependent* model ([8]) with linear topology (no document clustering). This model owes much to previous work in the text domain [34] and statistical machine translation [18, 19, 40].

The general idea, common with many models in this genre, is that image are generated from latent factors (concepts) which contribute both visual entities and words. The fact that visual entities and words come from the same source is what enables the model to link them. Because we train the models without knowing the correspondence, we need an assumption of how multiple draws from the pool of factors lead to the observed data with ambiguity. The dependent model is distinguished by assuming that multiple draws are first made to produce the observed image entities. The same group of factors is then sampled to produced the image words. Because words are generated conditioned on the observed image, we consider this to be a translation approach.

This approach will work with any characterization of image entities (e.g. regions with features). However, a key assumption is that image semantics is compositional, and thus each image typically needs to be described by multiple visual entities. Without compositionally, we would need to model all possible combinations of entities. For example, we would have to model tigers on grass, tigers in water, tigers on sand, and so on. Clearly, one tiger model should be reused when possible.

In what follows, we use feature vectors associated with image regions obtained using normalized cuts [45]. For each image region we compute a feature vector representing color, texture, size, position, shape [8], and color context [11]. As in earlier work, we will refer to region, together with its feature vector, as a *blob*. Our segmentations are limited to grouping pixels together with coherent color and texture, and thus should be considered very low level.

2.1 An exemplar multi-modal translation model

We model the joint probability of a particular blob, b, and a word w, as

$$P(w,b) = \sum_l P(w|l)P(b|l)P(l) \tag{1}$$

where l indexes over concepts, $P(l)$ is the concept prior, $P(w|l)$ is a frequency table, and $P(b|l)$ is a Gaussian distribution over features. We further assume a

diagonal covariance matrix (independent features) because fitting a full covariance is generally too difficult for a large number of features. This independence assumption is less troublesome because we only require conditional independence, given the concept. Intuitively, each concept generates some image regions according to the particular Gaussian distribution for that concept. Similarly, it generates one ore more words for the image according to a learned table of probabilities.

To go from the blob oriented expression (1) to one for an entire image, we assume that the observed blobs, B, yield a posterior probability, $P(l|B)$, which is proportional to the sum of $P(l|b)$. Words are then generated conditioned on the blobs from:

$$P(w|B) \propto \sum_{l} P(w|l)P(l|B) \tag{2}$$

where by assumption

$$P(l|B) \propto \sum_{b} P(l|b) \tag{3}$$

and Bayes rule is used to compute $P(l|b) \propto P(b|l)P(l)$.

Some manipulation [9] shows that this is equivalent to assuming that the word posterior for the image is proportional to the sum of the word posteriors for the regions:

$$P(w|B) \propto \sum_{b}^{N} P(w|b) \tag{4}$$

We limit the sum over blobs to the largest N blobs (in this work N is sixteen). While training, we also normalize the contributions of blobs and words to mitigate the effects of differing numbers of blobs and words in the various training images. The probability of the observed data, $W \cup B$, given the model, is thus:

$$P(W \cup B) = \prod_{b \in B} \left(\sum_{l} P(b|l)P(l) \right)^{\frac{max(N_b)}{N_b}} \prod_{w \in W} \left(\sum_{l} P(w|l)P(l|B) \right)^{\frac{max(N_w)}{N_w}} \tag{5}$$

where $max(N_b)$ (similarly $max(N_w)$) is the maximum number of blobs (words) for any training set image, N_b (similarly N_w)is the number of blobs (words) for the particular image, and $P(l|B)$ is computed from (3).

Since we do not know which concept is responsible for which observed blobs and words in the training data, determining the maximum likelihood values for the model parameters ($P(w|l)$, $P(b|l)$, and $P(l)$) is not tractable. We thus estimate values for the parameters using expectation maximization (EM) [23], treating the hidden factors (concepts) responsible for the blobs and words as missing data.

The model generalizes well because it learns about image components. These components can occur in different configurations and still be recognized. For example, it is possible to learn about "sky" regions in images of tigers, and then

predict "sky" in giraffe images. Of course, predicting the word giraffe requires having giraffes in the training set.

3 Reducing correspondence ambiguity with adjectives

We assume that descriptive text of an image can be parsed into parts of speech with reasonable accuracy [16, 17]. We further assume that the nouns that adjectives bind to can be identified. Finally, in order to be useful, adjectives need to be *visual* relative to a set of features. Examples of visual adjectives include color words (e.g. "red"), and texture words (e.g. "furry"). We address pruning non-visual adjectives from our vocabulary in the next section (§4).

Under these assumptions, it should be clear that adjectives have the potential to help with correspondence disambiguation. If we are not (yet) able to link a red ball to a circular red region, but we have the binding "red ball", and we have a model for red, then we have evidence that "ball" should link to red image regions, and not other ones. We assume that if an adjective, a, binds to a noun, n, then:

$$P(n|b) \propto P(b|n) \propto P(b|a) \propto P_{adj}(a|b) \tag{6}$$

where P_{adj} indicates that we use an adjective model. It is conceivable to construct a process to jointly learns an adjective model and a noun model. However, it is simpler to compute an adjective model first, using (for example) (5) restricted to adjectives, and then use (6) as a prior probability for the nouns (6). That prior is than used with a noun model, such as(5) applied to nouns.

To test the hypothesis that visual adjectives can help reduce correspondence ambiguity in training data, we constructed a small data by labeling many of the nouns associated with 1900 Corel ^{TM} images with one of fifteen adjectives which were expected to have good visual properties (11 were color words). We then built a prediction model for the adjectives alone using the model reviewed above (§2.1). Thus we learned a model that could predict, to a certain extent, "red" for a red region. We then applied the adjective based posterior to get a noun prior via the linking of nouns with adjectives. We assumed that most of the probability mass for this prior should be distributed among the associated words for that image, but since the annotations often do not cover all blobs, we allowed 10% of the probability mass for words not in the annotation. We also build an instance of the same model (§2.1) for nouns. We then combined the evidence from the noun model and adjective model used to predict the nouns that they modify.

The results are much as one would expect. Some difficult to characterize nouns are relatively easy to label given this kind of additional, semi-supervisory, information. Almost invariably the labeling of the training data was improved by including the adjective information. Often it was a more reliable source of information than the noun model. This is likely partly due to the nature of our "toy" data set, which has more nouns associated with visual adjectives than would commonly be the case.

A main use of adjective information is to help label data that is not strongly correlated with simple visual features. A good example is the car image (Figure 1). Since cars come in all colors, learning to recognize cars as an object class by color is not possible. However, color can be used to identify a particular instance of a car. Several examples can further ensure that that cars are dissociated from a narrow range of colors. Given the identified examples, we are then in a better position to construct a car model.

The fact that we chose to learn the meaning of the adjectives from a small, weakly labeled data set, means that there were some labeling errors due to the imperfect adjective model. This could be improved by more data, or by adding some truly supervisory information. A second problem with our current system is that good labellings based on adjectives are often better than the combined result. We are currently pursuing better integration of the two sources of information.

4 Identifying visual words

The above proof of concept relied on having nouns associated with adjectives that had a good chance of being linked with our features. When we apply the methods discussed above to larger data sets with free form text, our vocabularies will gain many entries that have no chance to be linked with visual properties measured by a given feature set. It is thus reasonable to attempt to prune vocabularies in advance, removing words that do not have significant correlations with our features. While is is conceivable that our models can simply absorb these words without any ill effect, it is more likely that the noise created by words with no visual properties will be detrimental. At a minimum, the computation cost can be reduced by excluding such words.

We consider determining the visualness of a word based on a large external data set that is not necessarily the target data set. While the visualness of a word is somewhat relative to the data set, many words may not occur frequently enough in a particular data set that a clear distinction can be made. We want to keep words that might be subtly visual in our data set, and prune as many as we can that have little chance of being visual at all.

Thus our approach is to actively seek many images that might be relevant to each word under consideration, and determine how visual that word is in general. Fortunately, with web image search engines such as Google Image Search, finding a large number of images that have a fair chance of being relevant to a given word is relatively straightforward.

Having selected the images, we face a familiar problem. Even if a word is relevant to an image in general, it likely correlates with the features of only a small part of the image. We expect the bulk of any image to be irrelevant to the word. Hence to estimate whether a word correlates with image features, we need to estimate which parts of the image are relevant. Not surprisingly, this requires an iterative algorithm which alternates between determining an

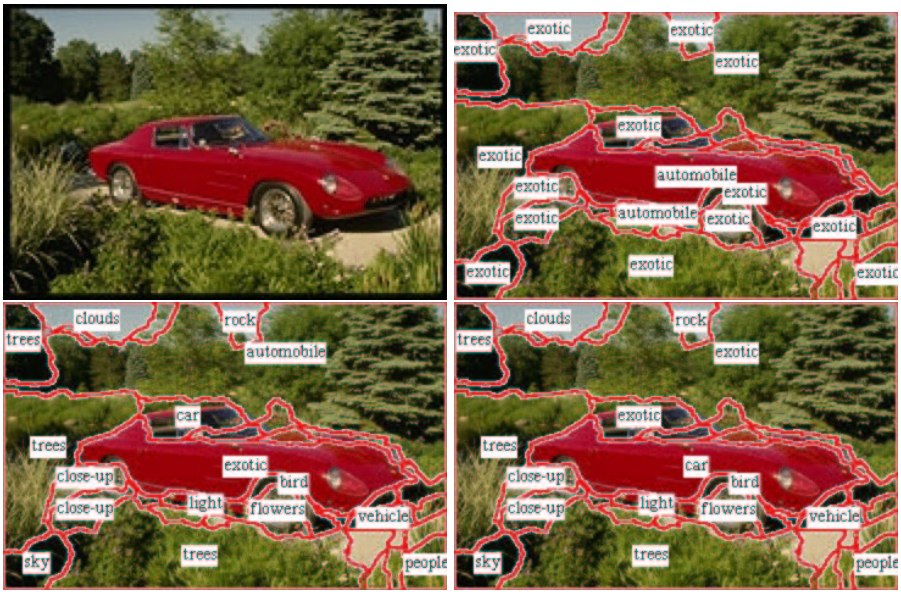

Fig. 1. Example of using adjectives to reduce correspondence ambiguity in training data. The upper left image is the original image containing a red car with a green vegetation backdrop. It is annotated with "red:automobile red:vehicle red:car exotic drago ferrari". The upper right image shows the nouns with maximal posterior probability for each region, based on the adjective model. Specifically, the red regions in the image are labeled by one of the nouns linked to "red". Regions that have low posterior given "red" are labeled by one of the words not linked to red (e.g. "exotic"). In this example, all words also refer to the car but this is not known at this point, and by exclusion, the non red regions get labeled with these other words. The bottom left image shows the labeling using the noun model alone, but with a strong prior (90%) on choosing among the associated words. This ensures that most of the words are good words for the image, but correspondence can be a problem, as is quite noticeable with "automobile" in a tree region. The bottom right image image shows the combined result. The correspondence has been enhanced by the adjective, promoting "car" to be the label for the body of the car. Several other words are reasonable, such as two instances of "trees". These words are not in the annotation, but they have sufficiently high posterior to overcome the prior that tends to restrict words to the ones from the annotation. Finally, it is clear that the word "exotic" is still ambiguous, due to being in the annotation of many car images with many backdrops, but having no clear visual properties.

appropriate characterization for the word, and determining which regions are relevant.

To implement this we prepare a large Gaussian mixture model for the regions of a large number of images. A concept is characterized as probability distribution over the mixture components. We iteratively estimate that distribution and the whether or not each image region is relevant to the concept. After sufficient iterations we compute the entropy of the distribution. If that distribution has low entropy, then we designate the word as visual. Otherwise, the process suggests that it is hard to distinguish the regions linked to the word with from a random selection of regions. In that case we consider that word not sufficiently visual, and prune it from the words that we try to link to image features. Some details follow.

4.1 Data gathering and pre-processing

For each concept (e.g. adjective) we use Google^{TM} image search to find several hundred images. As in the previous section, we simplify the data using low level segmentation. However, due to the volume of data, we segment images with JSEG [24] instead of normalized cuts, which is more expensive. For all processing that follows we used the same feature set described above.

4.2 Detecting regions associated with a concept

We process each concept in sequence. For each concept "X" we process the regions from the associated images, as well as an equal number of randomly selected other images, providing "non-X" regions. To obtain $P(X|r_i)$, which represents the probability that a region is associated with the concept "X" we use the following iterative process.

At first, we select "X" regions from the "X" images, and some "non-X" regions from the "non-X" images at random. We then fit a Gaussian mixture model for the image region features for both "X" and "non-X", and assign components of the mixture model according to the following formula:

$$p_j^X = \sum_{i=1}^{n_X} P(c_j|r_i^X, X) \tag{7}$$

$$= \sum_{i=1}^{n_X} P(X|c_j, r_i^X)P(c_j) \tag{8}$$

where c_j is the j-th component of the mixture model, n_X is the number of "X" regions, and r_i^X is the i-th "X" region.

The top m components in terms of p_j^X are regarded as the model of "X" and the rest are the model of "non-X". With these models of "X" and "non-X", we can compute $P(X|r_i)$ for all the regions which come from "X" images. Assuming

that $p1(X|r_i)$ is the output of the model of "X" and $p2(nonX|r_i)$ is the output of the model of "non-X", given r_i, we can obtain $P(X|r_i)$ as follows:

$$P(X|r_i) = \frac{p1(X|r_i)}{p1(X|r_i) + p2(nonX|r_i)} \tag{9}$$

For the next iteration, we select the top n regions regarding $P(X|r_i)$ as "X" regions and the top $n/2$ regions regarding $P(nonX|r_i)$ as "non-X" regions. Add $n/2$ regions randomly selected from "non-X" images to "non-X" regions. In this way, we mix newly estimated "non-X" regions and randomly selected regions from "non-X" images after the second iteration. We adopt mixing rather than using only newly estimated "non-X" regions based on the results of the preliminary experiments. After computing the entropy, we repeat estimation of the model of "X" and "non-X", and computation of $P(X|r_i)$.

4.3 Computing the entropy of concepts

We estimate the entropy of the image features of all the regions weighted by $P(X|x_i)$ with respect to a generic model for image regions. For this model we use a Gaussian mixture model (GMM) for fifty thousand randomly selected regions from all the images. To reduce the impact of initialization in the EM process, we average the results over k GMM's fit with different starting points.

The average probability of image features of "X" weighted by $P(X|x_i)$ with respect to the j-th component of the l-th generic base represented by the GMM is given by

$$P(X|c_j, l) = \frac{w_{j,l} \sum_{i=1}^{N_X} P(f_{X,i}; \theta_{j,l}) P(X|r_i)}{\sum_{i=1}^{N_X} P(X|r_i)} \tag{10}$$

where $f_{X,i}$ is the image feature of the i-th region of "X", $P(f_{X,i}; \theta_{j,l})$ is the generative probability of $f_{X,i}$ from the j-th component, $w_{j,l}$ is the weight of the j-th component of the l-th base, and N_X is the number of all the regions which come from "X" images,

The entropy for "X" is given by

$$E(X) = \frac{1}{k} \sum_{l=1}^{k} \sum_{j=1}^{N_{\text{base}}} -P(X|c_j, l) \log_2 P(X|c_j, l) \tag{11}$$

where N_{base} is the number of the components of the base (250 in our experiments), and k is number of GMM's with different starting points (5 in our experiments). We use this entropy as a measure of the visualness of a concept.

4.4 Experiments

We experimented with 150 adjectives which are the 150 most common adjectives used for indexing images in the Hemera Photo-Object collection. We used each of these adjectives as the search term for Google Image search. We used the first

250 web images returned. Thus the entire experiment considered nearly forty thousand images associated with adjectives.

We used 15 mixture components in (7). Because we expect adjectives to be associated with visual properties more directly than nouns, we simply use a single mixture component to model "X" (i.e., m=1).

Figure 2 shows "yellow" images after one iteration. In the figure, the regions with high probability $P(\text{yellow}|r_i)$ are labeled as "yellow", while the regions with high probability $P(\text{non\_yellow}|r_i)$ are labeled as "non-yellow". Figure 3 shows "yellow" images after five iterations. This indicates the iterative region selection worked well in case of "yellow".

Table 1 shows the 15 top adjectives and their image entropy. In this case, the entropy of "dark" is the lowest, so in this sense "dark" is the most "visual" adjective among the 150 adjectives under the condition we set in this experiment. Figure 4 shows some of the "dark" images. Most of the region labeled with "dark" are uniform black ones. Other highly-ranked adjectives, "senior" and "beautiful" include many human faces, and "visual", which, interestingly, are not photos but graphical images such as screen shots of Windows or Visual C. This suggests that addressing biases due to what images are common on the web may be helpful.

We provide the ranking of color adjectives in Table 1. They are relatively high, even though images from the Web included many irrelevant images. This suggests that our pruning approach is promising.

Notice that the method identifies many words which, at first glance, do not appear to be truly visual. A good example in our results is "professional" which is ranked relatively high. The connection is through the sampling bias for "professional sports" which yields low entropy because of a limited number of textures and backgrounds (e.g. fields and courts) that go with those images. It would seem to depend on the application as to whether these words are a liability. If the goal is to help image search, then such associations can be helpful. However, we have clearly not captured the essence of "professional", and thus for recognition we would hope that the ambiguity can be resolved in subsequent steps.

This is conceivable in many cases. In the "professional sports" case, if we assume relatively rich descriptions and sufficient data, then in the generative model above, words like "field" and "court" would *compete* with "professional" for probability. This can promote "professional" as a more general term that is less directly associated with local features.

Table 2 lists the 15 adjectives with lowest entropy among the 150 tested. In case of "religious" (Figure 5), which is ranked as 145-th, the region-adjective linking did not work well, and the entropy is thus relatively large. This reflects the fact that the image features of the regions included in "religious" images have no prominent tendency. Thus we can say that "religious" has no or only a few visual properties.

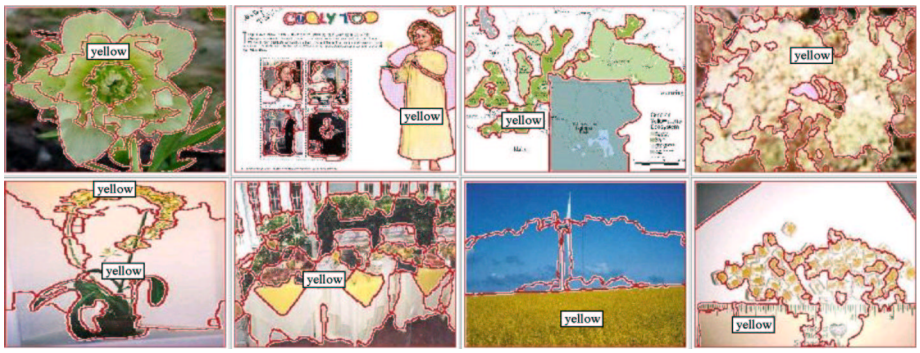

Fig. 2. "Yellow" regions after one iteration. At this stage many of the images do not have much yellow in them, and there are many labeling errors. For example, the flower in the top right image is green-blue, as is the region in the third image in the top row. The region marked yellow in the second image of the second row is white, whereas the two smaller, un-labeled, regions to either side are in fact yellow.

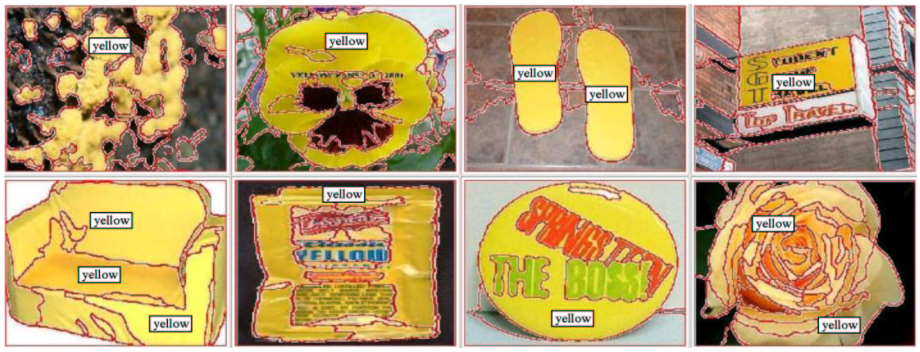

Fig. 3. "Yellow" regions after five iterations. These images all have significant yellow regions, and they are generally correctly labeled.

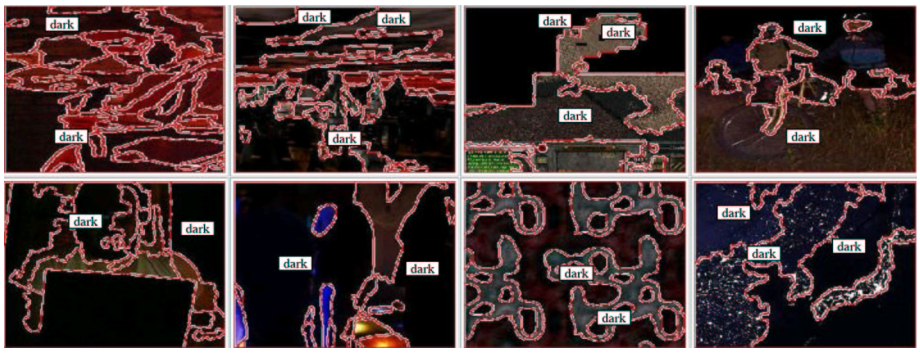

Fig. 4. "Dark" regions after five iterations.

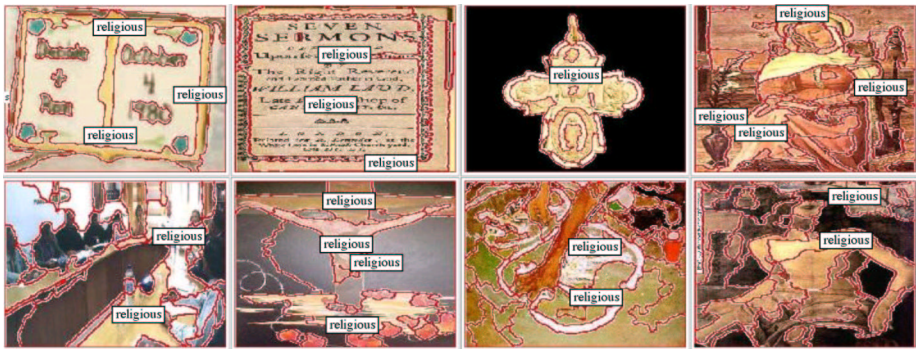

Fig. 5. "Religious" regions after five iterations.

Table 1. Words with the top 15 entropy rankings.

rank	adjective	entropy
1	dark	0.0118
2	senior	0.0166
3	beautiful	0.0178
4	visual	0.0222
5	rusted	0.0254
6	musical	0.0321
7	purple	0.0412
8	black	0.0443
9	ancient	0.0593
10	cute	0.0607
11	shiny	0.0643
12	scary	0.0653
13	professional	0.0785
14	stationary	0.1201
15	electric	0.1411

Table 2. Words with the bottom 15 entropy rankings.

rank	adjective	entropy
136	medical	2.5246
137	assorted	2.5279
138	large	2.5488
139	playful	2.5541
140	acoustic	2.5627
141	elderly	2.5677
142	angry	2.5942
143	sexy	2.6015
144	open	2.6122
145	religious	2.7242
146	dry	2.8531
147	male	2.8835
148	patriotic	3.0840
149	vintage	3.1296
150	mature	3.2265

Table 3. Rankings of color adjectives.

(color adjectives)		
7	purple	0.0412
8	black	0.0443
36	red	0.9762
39	blue	1.1289
46	yellow	1.2827

5 Using pictures to understand language

Links between visual features and words can also be exploited for understanding text and other documents. The idea is very simple and very familiar — illustrations can help clarify and enhance the meaning of documents. As an initial step in making this operational in an automatic setting, we have studied the problem of using images to help disambiguate word senses [14].

Words used in natural language are often ambiguous because language has evolved so that many words have several distinct meanings (senses). For example, the word "bank" can mean a financial institution or a step or edge as in "snow bank" or "river bank". Words which are spelled the same but have different meanings (polysemes) confound attempts to automatically understand natural language.

Because such words are very prevalent, determining the correct sense (word sense disambiguation) has been identified as an important problem in natural language processing research. As such, it has been studied by many researchers leading to a large body of work [7, 37, 51, 50, 32, 3, 2, 42, 43, 49].

Since the words are spelled the same, resolving their sense requires considering their context. A purely natural language based approach considers words near the one in question. Thus in the bank example, words like "financial" or "money" are strong hints that the financial institution sense is meant. Interestingly, despite much work, and a number of innovative ideas, doing significantly better than choosing the most common sense remains difficult [49].

To use our word prediction model for word sense disambiguation, we constrain the predicted words to be from the set of senses for the word being analyzed. In general, when word prediction is constrained to a narrow set of choices (such as possible senses), reasonable performance is possible. This is the key point. A very limited understanding of what is in the image can be helpful for sense disambiguation. All that is required is that the image is more likely to be associated with the correct sense, compared to a handful of others.

Associated images can help improve document retrieval. Invariably the senses of the words available in unstructured data are not sense disambiguated. Being able to automatically reduce the ambiguity should improve the quality of results.

Notice that in this scenario, we assume that the user is willing to indicate the query term sense. However, the general thrust of the method can take an implicit role. Specifically, even without sense information, retrieved documents can be organized on semantic lines for searching, browsing and relevance feedback based on a combination of words and visual features of associated images. To the extent that the later are linked to semantics based on training data, the associated images can help specify text semantics.

5.1 Predicting senses based on visual information

In the context of word sense disambiguation, our vocabulary is assumed to be sense disambiguated. Formally, we use an extended vocabulary S, which contains the senses of the words in a vocabulary W. Notationally, if the word *bank*

$\in W$ then $\{$*bank_1, bank_2, ...*$\} \in S$. Thus, every sense $s \in S$ is the sense of only one word $w \in W$. Once a model has been trained on S, we can use the annotation process to compute $P(s|B)$. Different than annotation, word sense disambiguation has the additional characteristic that we are trying to *only* distinguish between the senses, s, for a particular word, w, rather than produce a number of good choices from all of S, which is clearly more difficult.

Thus given a word, w, we assume that senses for all other words should not be predicted. Operationally we simply take the posterior probability over all the senses in our vocabulary, and set those not corresponding to w to zero. We then rescale the posterior so that it sums to one. This computation yields the probability of a word sense, s, given w, and the visual context, B, which we denote as $P(s|w, B)$.

5.2 Combining word prediction with text based word sense disambiguation

The quantity $P(s|w, B)$ can be used as is for word sense disambiguation, and we provide results for this strategy. It is also natural to combine it with text based methods, as it seems to provide an orthogonal source of information. Here we assume that a text based method can provide a second estimate of the probability $P(s|w, W)$ for the sense, s, for w, based on the observed words, W (the senses are not known a priori). We discuss our choice of $P(s|w, W)$ below (§5.3).

We assume that these two estimates are relatively independent, which gives the following simple expression for combining them:

$$P(s|w, B, W) \propto P(s|w, B)P(s|w, W). \tag{12}$$

5.3 Text based word sense disambiguation

The probability $P(s|w, W)$ in (12) is assumed to come from a traditional text based word sense disambiguation algorithm. We report results using the state of the art SMUaw algorithm [43]. This algorithm, and a recent derivative, SenseLearner [41]), have performed very well in word sense disambiguation challenges [26, 1]. We modified the SMUaw algorithm to give softer output so that it would work better with our approach ([14]).

5.4 ImCor

To develop and test methods for using images to disambiguate text, one requires a data set that has images linked to sense disambiguated text. As no such data was readily available, we developed a new corpus, ImCor with these properties. This data is available for research purposes [36].

To construct ImCor we linked images from the Corel^{TM} data set to passages from the already sense-attributed corpus, SemCor [43, 47, 33, 44]. SemCor, short for the WordNet Semantic Concordance [31], consists of 25% of the Brown corpus [30] files which have been fully tagged with part-of-speech and is sense

disambiguated. Since the SemCor files contain sizable text passages, we selected the relevant subset of a file to link with each image. Two participants carefully linked 1633 images with an overlap of 1/6 to verify consistency. We then automatically expanded the set to 20,153 image/text pairings by exploiting the semantic redundancy in the Corel^{TM} data, by linking images that shared two or more keywords with the manually linked images.

5.5 Experiments

To test our approach we created twenty different splits of ImCor into training and testing sets (90% training, 10% testing). Since there are a number of images which are used multiple times, we took care to ensure that all duplicate images were considered to be in either the training or testing sets for a given run. For each split, we then determined the vocabulary from the training data. First we removed stop words from the corpus. Then we eliminated word senses which occurred less than 20 times. If this produced images without words, they were removed, and the vocabulary was recomputed, iteratively, if needed. Typical vocabulary sizes were 3800 senses from about 3100 sense blind words.

We trained the word prediction model (§2) on the combined image sense data. We used the features described above for the 16 largest regions. If there were fewer than 16 regions, the we used all of them. We then applied the model to the test data to predict senses according to (4), by restricting word prediction to the sense for each word being processed as described above. We them combined visual and textual cues as described in §5.2.

We computed performance using *only* documents which have at least one ambiguous word. We used the performance of the empirical distribution of the training set for a baseline. Baseline performance on sense prediction was roughly 60%. This baseline provides a harsher standard than the simple "most common sense" method, as the empirical distribution gives the common sense for the em particular corpus.

In Table 4 provides the average absolute sense prediction scores over the 20 samples. More detailed results have been reported elsewhere [14]. The results of combining the two sources of information are very promising. The performance exceeds that of either method alone, which was what we were trying to achieve. On the large data set we were able to increase performance over the baseline by nearly 20% yielding nearly 80% absolute performance. In the small data set, the performance increase was more modest, yielding 5% improvement. We emphasize that our domain was constructed somewhat artificially to test our ideas, and that some of the improvement going from the small (seed) data set to the larger one is likely due to the system taking advantage of the structure of the Corel^{TM} data. However, even in the seed data case, where there was only limited training data (but the corpus was more pure), including image data produced a statistically significant improvement in word sense disambiguation performance.

Data set	Minimum sense count	Baseline	Text only using [43]	Image only	Combined (using (12))
Full	20	0.615	0.683	0.791	0.817
Seed	20	0.571	0.693	0.687	0.741

Table 4. Word sense prediction results. The first row is for the extended ImCor data set (20,153 text passages paired with images). The second row shows the result using the manually produced seed data set (1,633 pairs), even though the data is a bit sparse for our learning method. The numbers tabulated are the fraction of times the sense was correctly chosen. Every document processed has at least one ambiguous word. Some unambiguous can accompany those, and all algorithms score correctly on them by construction. All results are the average of 20 different splits of training and testing. The error, as estimated from the variance over the 20 test/training splits, is about 0.003 for the first row, and about 0.01 for the second row. tests.

6 Conclusion

Data with multiple modalities present great opportunities to learn semantics beyond what is possible considering the modes separately. In general, we will be more successful if we combine information from all available sources. We have presented several examples for doing this in the case of images with associated text and vice versa.

We have demonstrated how language structure can help reduce correspondence ambiguities in loosely labeled data. In particular, adjectives extracted from text can be help push loosely labeled data towards labeled data. Such an approach is important because many current methods for learning recognition rely on non-negligible quantities of data. Since labeled data is rare, but loosely labeled data is relatively easy to acquire, strategies for reducing the ambiguity of the labeling are clearly useful. Because these efforts are on a large scale, we have also studied the problem of how to prune words that are not visual given a feature set. Such pre-processing will be helpful for developing systems that learn for large scale data with free form text. In particular, the method addresses the problem that noise from non-visual words can overwhelm attempts to automatically learn the meaning of others that have more substantive links to features.

We have also summarized recent work on using images to help the understanding of natural language. In particular, correlations with visual attributes can help disambiguate word senses. Because the word prediction machinery is applied to merely choosing among the various senses of one word, visual information can be quite helpful, despite current limitations in image understanding.

We remark that it is also the limited number of choices that makes obtaining reasonable labeled data from loosely labeled image data reasonable. Here we only need to differentiate among the visual words associated with the images, which is generally a relatively small set compared to the entire vocabulary. Once the correspondence ambiguity has been reduced, we are then in a better position to

learn more sophisticated processes and models which are necessary for inference on novel data.

References

1. *Proceedings of Senseval-3: The Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, 2004.
2. E. Agirre and G. Rigau. Word sense disambiguation using conceptual density. In *Proceedings of COLING'96*, pages 16–22, Copenhagen, Denmark, 1996.
3. Eneko Agirre and German Rigau. A proposal for word sense disambiguation using conceptual distance. In *Proceedings of the 1st International Conference on Recent Advances in Natural Language Processing*, 1995.
4. R.A. Amar, D.R. Dooly, S.A. Goldman, and Q. Zhang. Multiple instance learning of real-valued data. In *18th Int. Conf. Machine Learning*, 2001.
5. Stuart Andrews, Thomas Hofmann, and Ioannis Tsochantaridis. Multiple instance learning with generalized support vector machines. In *AAAI*, 2002.
6. Stuart Andrews, Ioannis Tsochantaridis, and Thomas Hofmann. Support vector machines for multiple-instance learning. In *Advances in Neural Information Processing Systems, 15*, 2002.
7. Y. Bar-Hillel. The present status of automatic translation of languages. In Donald Booth and R.E. Meagher, editors, *Advances in Computers*, pages 91–163, New York, 1960. Academic Press.
8. Kobus Barnard, Pinar Duygulu, Nando de Freitas, David Forsyth, David Blei, and Michael I. Jordan. Matching words and pictures. *Journal of Machine Learning Research*, 3:1107–1135, 2003.
9. Kobus Barnard, Pinar Duygulu, and David Forsyth. Exploiting text and image feature co-occurrence statistics in large datasets. In Remco Veltkamp, editor, *Trends and Advances in Content-Based Image and Video Retrieval*. Springer, to appear.
10. Kobus Barnard, Pinar Duygulu, Nando de Freitas, David Forsyth, David Blei, and Michael I. Jordan. Matching words and pictures. *Journal of Machine Learning Research*, 3:1107–1135, 2003.
11. Kobus Barnard, Pinar Duygulu, K. G. Raghavendra, Prasad Gabbur, and David Forsyth. The effects of segmentation and feature choice in a translation model of object recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages II:675–682, 2003.
12. Kobus Barnard, Quanfu Fan, Ranjini Swaminathan, Anthony Hoogs, Roderic Collins, Pascale Rondot, and John Kaufhold. Evaluation of localized semantics: data, methodology, and experiments. Technical report, University of Arizona, 2005.
13. Kobus Barnard and David Forsyth. Learning the semantics of words and pictures. In *International Conference on Computer Vision*, pages II:408–415, 2001.
14. Kobus Barnard and Matthew Johnson. Word sense disambiguation with pictures. *Artificial Intelligence*, 167:13–30, 2005.
15. Alexander C. Berg, Tamara L. Berg, and Jitendra Malik. Shape matching and object recognition using low distortion correspondence. In *CVPR*, 2005.
16. Eric Brill. A simple rule-based part of speech tagger. In *Third Conference on Applied Natural Language Processing*. ACL, 1992.
17. Eric Brill. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics*, 21(4):543–565, 1995.

18. Peter F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Fedrick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. A statistical approach to machine translation. *Computational Linguistics*, 16:79–85, 1990.
19. Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. The mathematics of machine translation: parameter estimation. *Computational Linguistics*, 19(10):263–311, 1993.
20. Peter Carbonetto, Nando de Freitas, and Kobus Barnard. A statistical model for general contextual object recognition. In *European Conference on Computer Vision*, pages I:350–362, 2004.
21. Chad Carson, Megan Thomas, Serge Belongie, Joseph M. Hellerstein, and Jitendra Malik. Blobworld: A system for region-based image indexing and retrieval. In *Third International Conference on Visual Information Systems*. Springer, 1999.
22. M. La Cascia, S. Sethi, and S. Sclaroff. Combining textual and visual cues for content based image retrieval on the web. In *IEEE Workshop on Content Based Access of Image and Video Libraries*, pages 24–28, 1998.
23. A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.
24. Y. Deng and B. S. Manjunath. Unsupervised segmentation of color-texture regions in images and video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(8):800–810, 2001.
25. Pinar Duygulu, Kobus Barnard, J.F.G de Freitas, and D.A. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *The Seventh European Conference on Computer Vision*, pages IV:97–112, 2002.
26. Phil Edmonds and Adam Kilgarriff, editors. *Journal of Natural Language Engineering*, volume 9, January 2003.
27. L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. In *Workshop on Generative-Model Based Vision*, 2004.
28. S.L. Feng, R. Manmatha, and Victor Lavrenko. Multiple bernoulli relevance models for image and video annotation. In *Proceedings of CVPR'04*, volume 2, pages 1002–1009, 2004.
29. R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2003.
30. W. Nelson Francis and Henry Kučera. *Frequency Analysis of English Usage. Lexicon and Grammar*. Houghton Mifflin, 1981.
31. Miller G., Leacock C., Randee T., and Bunker R. A semantic concordance. In *Procedings of the 3rd DARPA Workshop on Human Language Technology*, pages 303–308, 1993.
32. W. Gale, K. Church, and D. Yarowsky. One sense per discourse. In *DARPA Workshop on Speech and Natural Language*, pages 233–237, 1992.
33. Julio Gonzalo, Felisa Verdejo, Irina Chugur, and Juan Cigarran. Indexing with wordnet synsets can improve text retrieval. In *Proceedings of the COLING/ACL '98 Workshop on Usage of WordNet for NLP*, pages 38–44, Montreal, Canada, 1998.
34. Thomas Hofmann and Jan Puzicha. Statistical models for co-occurrence data. Technical report, Massachusetts Institute of Technology, 1998.
35. J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *SIGIR*, pages 119–126, 2003.

36. Matthew Johnson and Kobus Barnard. *ImCor: A linking of SemCor sense disambiguated text to corel image data.* http://kobus.ca/research/data/index.html, 2004.
37. A. Kaplan. An experimental study of ambiguity in context, 1950.
38. O. Maron and T. Lozano-Perez. A framework for multiple-instance learning. In *Neural Information Processing Systems.* MIT Press, 1998.
39. O. Maron and A.L. Ratan. Multiple-instance learning for natural scene classification. In *The Fifteenth International Conference on Machine Learning*, 1998.
40. Dan Melamed. *Empirical methods for exploiting parallel texts.* MIT Press, Cambridge, Massachusetts, 2001.
41. Rada Mihalcea and Ehsanul Faruque. Senselearner: Minimally supervised word sense disambiguation for all words in open text. In *Proceedings of ACL/SIGLEX Senseval-3*, Barcelona, Spain, July 2004.
42. Rada Mihalcea and Dan Moldovan. Word sense disambiguation based on semantic density. In *Proceedings of the COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems*, Montreal, Canada, August 1998.
43. Rada Mihalcea and Dan Moldovan. An iterative approach to word sense disambiguation. In *Proceedings of Florida Artificial Intelligence Research Society Conference (FLAIRS 2000)*, pages 219–223, Orlando, FL, May 2000.
44. Palomar M. Montoyo A. and Rigau G. Wordnet enrichment with classification systems. In *Proceedings of NAACL Workshop 'WordNet and Other Lexical Resources: Applications, Extensions and Customizations'*, pages 101–106, Carnegie Mellon University, Pittsburgh, USA, 2001.
45. J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(9):888–905, 2000.
46. Nikhil V Shirahatti and Kobus Barnard. Evaluating image retrieval. In *Proceedings of CVPR'05*, volume 1, pages 955–961, 2005.
47. Jiri Stetina, Sadao Kurohashi, and Makoto Nagao. General word sense disambiguation method based on A full sentential context. In Sanda Harabagiu, editor, *Use of WordNet in Natural Language Processing Systems: Proceedings of the Conference*, pages 1–8. Association for Computational Linguistics, Somerset, New Jersey, 1998.
48. A. Torralba, K. P. Murphy, and W. T. Freeman. Sharing features: efficient boosting procedures for multiclass object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages II:762–769, 2004.
49. Jonathan Traupman and Robert Wilensky. Experiments in improving unsupervised word sense disambiguation. Technical report, University of California at Berkeley, 2003.
50. David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *33rd Conference on Applied Natural Language Processing.* ACL, 1995.
51. V. Yngve. Syntax and the problem of multiple meaning. In W. Locke and D. Booth, editors, *Machine Translation of Languages*, pages 208–226, New York, 1955. Wiley.
52. Q. Zhang and S.A. Goldman. Em-dd:an improved multiple-instance learning technique. In *Neural Information Processing Systems*, 2001.
53. Q. Zhang, S.A. Goldman, W. Yu, and J.E. Fritts. Content-based image retrieval using multiple-instance learning. In *19th Int. Conf. Machine Learning*, 2001.