Multiple Bernoulli Relevance Models for Image and Video Annotation

S. L. Feng, R. Manmatha and V. Lavrenko *
Multimedia Indexing and Retrieval Group
Center for Intelligent Information Retrieval
University of Massachusetts
Amherst, MA, 01003

Abstract

Retrieving images in response to textual queries requires some knowledge of the semantics of the picture. Here, we show how we can do both automatic image annotation and retrieval (using one word queries) from images and videos using a multiple Bernoulli relevance model. The model assumes that a training set of images or videos along with keyword annotations is provided. Multiple keywords are provided for an image and the specific correspondence between a keyword and an image is not provided. Each image is partitioned into a set of rectangular regions and a real-valued feature vector is computed over these regions. The relevance model is a joint probability distribution of the word annotations and the image feature vectors and is computed using the training set. The word probabilities are estimated using a multiple Bernoulli model and the image feature probabilities using a non-parametric kernel density estimate. The model is then used to annotate images in a test set. We show experiments on both images from a standard Corel data set and a set of video key frames from NIST's Video Trec. Comparative experiments show that the model performs better than a model based on estimating word probabilities using the popular multinomial distribution. The results also show that our model significantly outperforms previously reported results on the task of image and video annotation.

1. Introduction

Searching and finding large numbers of images and videos from a database is a challenging problem. The conventional approach to this problem is to search on image attributes like color and texture. Such approaches suffer from a number of problems. They do not really capture the semantics of the problem well and they often require people to pose image queries using color or texture which is difficult for most people to do. The traditional "low-tech" solution to this problem practiced by librarians is to annotate each image manually with keywords or captions and then search on those captions or keywords using a conventional text search engine. The rationale here is that the keywords capture the semantic content of the image and help in retrieving the images. This technique is also used by television news organizations to retrieve file footage from their videos. While "low-tech", such techniques allow text queries and are successful in finding the relevant pictures. The main disadvantage with manual annotations is the cost and difficulty of scaling it to large numbers of images.

Automatically annotating images/videos would solve this problem while still retaining the advantages of a semantic search. Here, we propose approaches to automatically annotating and retrieving images/videos by learning a statistical generative model called a relevance model using a set of annotated training images. The images are partitioned into rectangles and features are computed over these rectangles. We then learn a joint probability model for (continuous) image features and words called a relevance model and use this model to annotate test images which we have not seen. Words are modeled using a multiple Bernoulli process and images modeled using a kernel density estimate.

We test this model using a Corel dataset provided by [5] and show that it outperforms previously reported results on other models. It performs 4 times better than a model based on machine translation [5] and better than one which models word probabilities using a multinomial to represent words. Existing annotation models [5, 3, 7, 8] by analogy with the text retrieval world have used the multinomial distribution to model annotation words. We believe that annotation text has very different characteristics than full text in documents and hence a Bernoulli distribution is more appropriate.

In image/video annotation, a multinomial would split the probability mass between multiple words. For example, if an image was annotated with "person, grass", with perfect annotation the probability for each word would be equal to

*This work was supported in part by the Center for Intelligent Information Retrieval and in part by the National Science Foundation under grant number IIS-9909073 and in part by SPAWARSYSCEN-SD grant number N66001-02-1-8903. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsor.

0.5. On the other hand another image which has just one annotation “person” would have a probability of 1.0 with perfect annotation. If we want to find images of people, when rank ordering these images by probability the second image would be preferred to the first although there is no reason for preferring one image over another. The problem can be made much worse when the annotation lengths for different images differ substantially. A similar effect occurs when annotations are hierarchical. For example, let one image be annotated “face, male_face, Bill_Clinton” and a second image be annotated with just “face”. The probability mass would be split three ways (0.33 each) in the first case while in the second image “face” would have a probability of 1. Again the second image would be preferred for the query “face”, although there is no reason for preferring one over the other. The Bernoulli model avoids this problem by making decisions about each annotation independent of the other words. Thus, in all the above examples, each of the words would have a probability of 1 (assuming perfect annotation).

It has been argued [14] that the Corel dataset is much easier to annotate and retrieve and does not really capture the difficulties inherent in more challenging (real) datasets like the news videos in Trec Video [12] We therefore, experimented with a subset of news videos (ABC, CNN) from the Trec Video dataset. We show that in fact we obtain comparable or even better performance (depending on the task) on this dataset and that again the Bernoulli model outperforms a multinomial model.

The specific contributions of this work include:

1. A probabilistic generative model which uses a Bernoulli process to generate words and kernel density estimate to generate image features. This model simultaneously learns the joint probabilities of associating words with image features using a training set of images with keywords and then generates multiple probabilistic annotations for each image.

2. Significant improvements in annotation performance over a number of other models on both a standard Corel dataset and a real word news video dataset.

3. Large improvements in annotation performance by using a rectangular grid instead of regions obtained using a segmentation algorithm (see [4] for a related result).

4. Substantial improvements in retrieval performance on one word queries over a multinomial model.

The focus of this paper is on models and not on features. We use features similar to those used in [5, 3]

The rest of this paper is organized as follows. We first discuss the multiple Bernoulli relevance model and its relation to the multinomial relevance model. This is followed by a discussion of related work in this area. The next section describes the datasets and the results obtained. Finally, we conclude the paper.

2 Multiple-Bernoulli Relevance Model

In this section we describe a statistical model for automatic annotation of images and video frames. Our model is called *Multiple-Bernoulli Relevance Model* (MBRM) and is based on the Continuous-space Relevance Model (CRM) proposed by [8]. CRM has proved to be very successful on the tasks of automatic image annotation and retrieval. In the rest of this section we discuss two shortcomings of the CRM in the video domain and propose a possible way of addressing these shortcomings. We then provide a formal description of our model as a generative process and complete the section with a brief discussion of estimation details.

2.1 Relation of MBRM and CRM

CRM[8] is a probabilistic model for image annotation and retrieval. The basic idea behind CRM is to reduce an image to a set of real-valued feature vectors, and then model the joint probability of observing feature vectors with possible annotation words. The feature vectors in [8] are based on automatic segmentation[10] of the target image into regions and are modeled using a kernel-based probability density function. The annotation words are modeled with a multinomial distribution. The joint distribution in [8] of words and feature vectors relies on a doubly non-parametric approach, where expectations are computed over each annotated image in the training set.

We believe the CRM model makes two assumptions that make it ill-suited for annotations in the image/video domain.

1. **Segmentation:** The CRM relies on automatic segmentation of the image into semantically-coherent regions. While the CRM does not make any assumptions about correspondence of annotation words to image regions, the overall annotation performance is strongly affected by the quality of segmentation. In addition, automatic segmentation is a rather expensive process that is poorly suited for large-scale video datasets.

2. **Multinomial:** CRM assumes that annotation words for any given image follow a multinomial distribution. This is a reasonable assumption in the Corel[5] dataset, where all annotations are approximately equal in length and words reflect *prominence* of objects in the image. However, in our video dataset[12] individual

frames have hierarchical annotations which do not follow the multinomial distribution. The length of the annotations also varies widely for different video frames. Furthermore, video annotations focus on *presence* of an object in a frame, rather than its *prominence*.

In the next two subsections we show how we can improve results by modifying these assumptions.

2.1.1 Rectangular image regions

In the current model, rather than attempting segmentation, we impose a fixed-size rectangular grid on each image. The image is then represented as a set of *tiles*. Using a grid provides a number of advantages. First, there is a very significant reduction in the computational time required for the model. Second, each image now contains a fixed number of regions, which simplifies parameter estimation. Finally, using a grid makes it somewhat easier to incorporate context into the model. For example, relative position could greatly aid in distinguishing adjacent tiles of water and sky. To evaluate the effect of using rectangular regions versus segmentation, we ran experiments with the CRM model but with rectangular regions as input - we call this CRM-Rectangles. The experiments in Section 4 show that this alone improves the mean per-word precision by about 38% - a substantial improvement in performance. We believe this is because segmentation is done on a per image basis. The CRM model cannot undo any problems that occur with segmentation. However, using a rectangular grid (with more regions than produced by the segmentation) allows the model to learn using a much larger set of training images what the correct association of words and image regions should be.

2.1.2 Multiple-Bernoulli word model

Another major contribution of the current model over the CRM is in our use of the multiple-Bernoulli distribution for modeling image annotations. In this section we highlight the differences between the multiple-Bernoulli and the multinomial model, and articulate why we believe that multiple-Bernoulli is a better alternative.

The multinomial model is meant to reflect the prominence of words in a given annotation. The event space of the model is the set of all *strings* over a given vocabulary, and consequently words can appear multiple times in the annotation. In addition, the probability mass is shared by all words in the vocabulary, and during the estimation process the words compete for this probability mass. As a result, an image I_1 annotated with a single word "face" will assign all probability mass to that word, so $P(\text{face}|I_1) = 1$. At the same time, an image I_2 annotated with two words "face" and "person" will split the probability mass, so $P(\text{face}|I_2) = \frac{1}{2}$. Thus the multinomial distribution models *prominence* of a word in the annotation, favoring single words, or words that occur multiple times in an annotation.

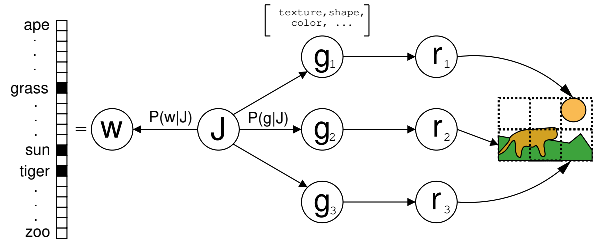


Figure 1: MBRM viewed as a generative process. The annotation w is a binary vector sampled from the underlying multiple-Bernoulli model. The image is produced by first sampling a set of feature vectors $\{g_1 ... g_n\}$, and then generating image regions $\{r_1 ... r_n\}$ from the feature vectors. Resulting regions are tiled to form the image.

Arguably, both images I_1 and I_2 contain a face, so the probability of "face" should be equal. This can be modeled by a multiple-Bernoulli model, which explicitly focuses on *presence* or *absence* of words in the annotation, rather than on their prominence. The event space of the multiple-Bernoulli model is the set of all *subsets* of a given vocabulary. Each subset can be represented as a binary occurrence vector in $\{0, 1\}^V$. Individual components of the vector are assumed to be independent and identically (Bernoulli-) distributed given the particular image.

In our dataset, image annotations are hierarchical and have greatly varying length. No word is ever used more than once in any given annotation, so modeling word frequency is pointless. Finally, words are assigned to the annotation based on merely the presence of an object in a frame, not on its prominence. We believe that a Bernoulli model provides a much closer match for this environment. Our hypothesis is supported by experimental results which will be discussed in section 4.

2.2 MBRM as a generative model.

Let $\mathcal{V}$ denote the annotation vocabulary, $\mathcal{T}$ denote the training set of annotated images, and let J be an element of $\mathcal{T}$. According to the previous section J is represented as a set of image regions $\mathbf{r}_J = \{r_1 ... r_n\}$ along with the corresponding annotation $\mathbf{w}_J \in \{0, 1\}^{\mathcal{V}}$. We assume that the process that generated J is based on two distinct probability distributions. First, we assume that the set of annotation words $\mathbf{w}_J$ is a result of $|\mathcal{V}|$ independent samples from every component of some underlying multiple-Bernoulli distribution $P_{\mathcal{V}}(\cdot|J)$. Second, for each image region r we sample a real-valued feature vector g of dimension k. The feature vector is sampled from some underlying multi-variate density function $P_{\mathcal{G}}(\cdot|J)$. Finally, the rectangular region r is produced according to some unknown distribution conditioned on g. We make no attempt to model the process of generating r from g. The resulting regions $r_1 \ldots r_n$ are tiled to form the image.

Now let $\mathbf{r}_A = \{g_1 \ldots g_{n_A}\}$ denote the feature vectors of some image A, which is not in the training set $\mathcal{T}$. Similarly, let $\mathbf{w}_B$ be some arbitrary subset of $\mathcal{V}$. We would like to model $P(\mathbf{r}_A, \mathbf{w}_B)$, the joint probability of observing an image defined by $\mathbf{r}_A$ together with annotation words $\mathbf{w}_B$. We hypothesize that the observation $\{\mathbf{r}_A, \mathbf{w}_B\}$ came from the same process that generated one of the images J^* in the training set $\mathcal{T}$. However, we don't know which process that was, and so we compute an expectation over all images $J \in \mathcal{T}$. The overall process for jointly generating $\mathbf{w}_B$ and $\mathbf{r}_A$ is as follows:

1. Pick a training image $J \in \mathcal{T}$ with probability $P_{\mathcal{T}}(J)$
2. Sample $\mathbf{w}_B$ from a multiple-Bernoulli model $P_{\mathcal{V}}(\cdot|J)$.
3. For $a = 1 \ldots n_A$:
    (a) Sample a generator vector g_a from the probability density $P_{\mathcal{G}}(\cdot|J)$.

Figure 1 shows a graphical dependency diagram for the generative process outlined above. We show the process of generating a simple image consisting of three regions and a corresponding 3-word annotation. Note that the number of words in the annotation n_B does not have to be the same as the number of image regions n_A. Formally, the probability of a joint observation $\{\mathbf{r}_A, \mathbf{w}_B\}$ is given by:

$$P(\mathbf{r}_A, \mathbf{w}_B) = \sum_{J \in \mathcal{T}} \left\{ P_{\mathcal{T}}(J) \prod_{a=1}^{n_A} P_{\mathcal{G}}(g_a|J) \times \right. \\ \left. \times \prod_{v \in \mathbf{w}_B} P_{\mathcal{V}}(v|J) \prod_{v \notin \mathbf{w}_B} (1 - P_{\mathcal{V}}(v|J)) \right\} \quad (1)$$

Equation (1) makes it evident how we can use MBRM for annotating new images or video frames. Given a new (un-annotated) image we can split it into regions $\mathbf{r}_A$, compute feature vectors $g_1 \ldots g_n$ for each region and then use equation 1 to determine what subset of vocabulary $\mathbf{w}^*$ is most likely to co-occur with the set of feature vectors:

$$\mathbf{w}^* = \arg \max_{\mathbf{w} \in \{0,1\}^{\mathcal{V}}} \frac{P(\mathbf{r}_A, \mathbf{w})}{P(\mathbf{r}_A)} \quad (2)$$

In practice we only consider subsets of a fixed size (5 words). One can show that the maximization in equation (2) can be done very efficiently because of the factored nature of the Bernoulli component. Essentially it can be shown that the equations may be simplified so that $P(w_i|J)$ may be computed independently for each word. This simplification arises because each word occurs at most once as the caption of an image. Space constraints preclude us from providing the proof.

2.3 Estimating Parameters of the Model

In this section we will discuss simple but effective estimation techniques for the three components of the model: $P_{\mathcal{T}}$, $P_{\mathcal{V}}$ and $P_{\mathcal{G}}$. $P_{\mathcal{T}}(J)$ is the probability of selecting the underlying model of image J to generate some new observation $\mathbf{r}, \mathbf{w}$. In the absence of any task knowledge we use a uniform prior $P_{\mathcal{T}}(J) = 1/N_{\mathcal{T}}$, where $N_{\mathcal{T}}$ is the size of the training set.

$P_{\mathcal{G}}(\cdot|J)$ is a density function responsible for generating the feature vectors $g_1 \ldots g_n$, which are later mapped to image regions $\mathbf{r}_J$ according to $P_{\mathcal{R}}$. We use a non-parametric kernel-based density estimate for the distribution $P_{\mathcal{G}}$. Assuming $g_J = \{g_1 \ldots g_n\}$ to be the set of regions of image J we estimate:

$$P_{\mathcal{G}}(g|J) = \frac{1}{n} \sum_{i=1}^{n} \frac{\exp\left\{-(g - g_i))^{\top} \Sigma^{-1} (g - g_i))\right\}}{\sqrt{2^k \pi^k |\Sigma|}} \quad (3)$$

Equation (3) arises out of placing a Gaussian kernel over the feature vector g_i of every region of image J. Each kernel is parametrized by the feature covariance matrix Σ. As a matter of convenience we assumed $\Sigma = \beta \cdot I$, where I is the identity matrix. β plays the role of kernel *bandwidth*: it determines the smoothness of $P_{\mathcal{G}}$ around the support point g_i. The value of β is selected empirically on a held-out portion of the training set $\mathcal{T}$.

$P_{\mathcal{V}}(v|J)$ is the v'th component of the multiple-Bernoulli distribution that is assumed to have generated the annotation $\mathbf{w}_J$ of image $J \in \mathcal{T}$. The Bayes estimate using a beta prior (conjugate to a Bernoulli) for each word is given by:

$$P_{\mathcal{V}}(v|J) = \frac{\mu \, \delta_{v,J} + N_v}{\mu + N} \quad (4)$$

here μ is a smoothing parameter estimated using the training and validation set, $\delta_{v,J} = 1$ if the word v occurs in the annotation of image J and zero otherwise. N_v is the number of training images that contain v in the annotation and N is the total number of training images.

3 Related Work

Our model differs from traditional object recognition approaches in a number of ways (for example [9, 13, 1, 6, 4, 11]. Such approaches require a separate model to be trained for each object to be recognized That is, even though the form of the statistical model may be the same, learning two different objects like a car and a person requires two separate training runs (one for each object). Each training run requires positive and negative examples for that particular object. On the other hand, in the relevance model approach described here all the annotation words are learned at the same time - each training image usually has many annotations. While some of the newer object recognition techniques [6] do not require training examples of the objects to be cut out of the background, they still seem to require one object in each image. Our model on the other hand can handle multiple objects in the same training image and can also ascribe annotations to the backgrounds like sky and grass. Unlike the more traditional object recognition techniques we label the entire picture and not specific image regions in a picture. This is as a librarian's manual annotation shows more than sufficient for tasks like retrieving images from a large database. The joint probability model that we propose takes context into account i.e. from training images it learns that an elephant is more likely to be associated with grass and sky and less likely to be associated with buildings and hence if there are image regions associated with grass, this increases the probability of recognizing the object as an elephant. Traditional object recognition models do not do this.

The model described here is closest in spirit to the annotation models proposed by [5, 3, 7, 8, 2]. Duygulu *et al* [5] proposed to describe images using a vocabulary of blobs. First, regions are created using a segmentation algorithm like normalized cuts. For each region, features are computed and then blobs are generated by clustering the image features for these regions across images. Each image is generated by using a certain number of these blobs. Their *Translation Model* applies one of the classical statistical machine translation models to translate from the set of keywords of an image to the set of blobs forming the image.

On the surface, MBRM appears to be similar to one of the intermediate models considered by Blei and Jordan [3]. Specifically, their *GM-mixture* model employs a similar dependence structure among the random variables involved. However, the topological structure of MBRM is quite different from the one employed by [3]. GM-mixture assumes a low-dimensional topology, leading to a fully-parametric model where 200 or so "latent aspects" are estimated using the EM algorithm. To contrast that, MBRM makes no assumptions about the topological structure, and leads to a doubly non-parametric approach, where expectations are computed over every individual point in the training set. In addition they model words using a multinomial process. Blei and Jordan used a different subset of the Corel dataset and hence it is difficult to make a direct quantitative comparison with their models.

MBRM is also related to the cross-media relevance model (CMRM) [7], which is also doubly non-parametric. There are three significant differences between MBRM and CMRM. First, CMRM is a discrete model and cannot take advantage of continuous features. In order to use CMRM for image annotation we have to quantize continuous feature vectors into a discrete vocabulary (similarly to the translation [5] models). MBRM, on the other hand, directly models continuous features. The second difference is that CMRM relies on *clustering* of the feature vectors into *blobs*. Annotation quality of the CMRM is very sensitive to clustering errors, and depends on being able to a-priori select the right cluster granularity: too many clusters will result in extreme sparseness of the space, while too few will lead us to confuse different objects in the images. MBRM does not rely on clustering and consequently does not suffer from the granularity issues. Finally, CMRM also models words using a multinomial process.

We would like to stress that the difference between MBRM and previously discussed models is not merely conceptual. In section 4 we will show that MBRM performs significantly better than all previously proposed models on the tasks of image annotation and retrieval. To ensure a fair comparison, we show results on exactly the same data set and similar feature representations as used in [5, 7, 8].

4. Experimental Results

We tested the algorithms using two different datasets, the Corel data set from Duygulu et al [5] and a set of video key frames from NIST's Video Trec [12]. To provide a meaningful comparison between MBRM and CRM-Rectangles, we do comparative experiments using the same set of features extracted from the same set of rectangular grids. For the Corel dataset we also compare the results with those of Duygulu et al and the CRM model.

4.1. Datasets and Feature sets

The Corel data set consists of 5000 images from 50 Corel Stock Photo cds. [1] Each cd includes 100 images on the same topic, and each image is also associated with 1-5 keywords. Overall there are 371 keywords in the dataset. In experiments, we divided this dataset into 3 parts: a training set of 4000 images, a validation set of 500 images and a test set of 500 images. The validation set is used to find model parameters. After finding the parameters, we merged the 4000 training set and 500 validation set to form a new training set. This corresponds to the training set of 4500 images and the test set of 500 images used by Duygulu *et al* [5].

We used a subset of NIST's Video Trec dataset (for computational reasons we did not use the entire data set). The

[1] We thank Kobus Barnard for making the Corel dataset available at http://www.cs.arizona.edu/people/kobus/research/data/eccv_2002

Models	Translation	CRM	CRM-Rectangles	MBRM
#words with recall ≥ 0	49	107	119	122
Results on 49 best words, as in[5, 7]				
Mean Per-word Recall	0.34	0.70	0.75	0.78
Mean Per-word Precision	0.20	0.59	0.72	0.74
Results on all 260 words				
Mean Per-word Recall	0.04	0.19	0.23	0.25
Mean Per-word Precision	0.06	0.16	0.22	0.24

Table 1: Performance comparison on the task of automatic image annotation on the Corel dataset. CRM and CRM-Rectangles are essentially the same model but the former uses regions produced by a segmentation algorithm while the latter uses a grid. Note that using a grid improves performance. MBRM performs best beating even CRM-Rectangles by a small amount.

data set consists of 12 mpeg files, each of which is a 30-minutes video section of CNN or ABC news and advertisements. 5200 key frames were extracted and provided by NIST for this dataset. The participants in TREC annotated a portion of the videos. The word vocabulary for human annotation is represented as a hierarchical tree with each annotation word as a node, which means many key frames are annotated hierarchically, e.g. a key frame can be assigned a set of words like ”face, male_face, male_news_subject”. This means that the annotation length for key frames can vary widely. There are 137 keywords in the whole dataset after we ignore all the audio annotations. We randomly divide the dataset into a training set (1735 key frames), a validation set (1735 key frames) and a test set (1730 key frames). As for the Corel set, the validation set is used to find system parameters, and then merged into the training set after we find the parameters.

Every image in these two sets is partitioned into rectangular grids, and a feature vector is then calculated for every grid region. The number of rectangles is empirically selected (using the training and validation sets) and is 24 for the Corel set, and 35 for the video dataset set. There are 30 features: 18 color features (including region color average, standard deviation and skewness) and 12 texture features (Gabor energy computed over 3 scales and 4 orientations).

4.2. Results of Automatic Image Annotation

In this section we evaluate the performance of our MBRM on automatic image annotation. Given an un-annotated image or key frame, we can calculate the generative probability of every candidate word in the vocabulary conditioned on it. For the Corel set, we take the top 5 words (according to probability) as automatic annotation of that image. For the video set, we take the top 6 (the average length of human annotations over all key frames) words. Figure 2 shows examples of the automatic annotations obtained using the CRM-Rectangles and MBRM models on the TREC Video. These results are obtained on the same dataset with identical preprocessing, features and training sets.

The first evaluation on annotation is done as in [5, 7, 8] using recall and precision calculated for every word in the test set. For this part of the process we do not use the actual rankings. Let A be the number of images automatically annotated with a given word, B the number of images correctly annotated with that word. C is the number of images having that word in ground-truth annotation. Then $recall = \frac{B}{C}$, and $precision = \frac{B}{A}$. To evaluate the system performance, recall and precision values are averaged over the testing words. The first set of results are shown for the Corel dataset in Table 1. Results are reported for all (260) words in the test set. They are also reported for the top 49 annotations to make a direct comparison with [5]. The three relevance model approaches are clearly much better than the translation model approach in [5] with MBRM outperforming all other models (4 times better than the translation model). CRM-Rectangles and CRM are identical except for the fact that CRM-Rectangles uses regions partitioned into rectangles while the regions in the CRM model are obtained using normalized cuts segmentation. As the results show this improves the performance significantly (almost 38% improvement in precision). Segmentation is a difficult error prone process in computer vision. The segmentation is done on a per image basis and hence there is some chance of combining semantically distinct regions together. Since the probabilistic model deals with regions as entities, it cannot undo segmentation errors (if for example two distinct image regions are combined together in the segmentation). However, if we start from a rectangular partition, the probabilistic model which learns from multiple training images has a better chance of associating the rectangular regions with the correct words. We believe that this accounts for the better performance using a rectangular partition.

Table 2 compares the annotation performance of CRM-Rectangles and MBRM and we see that the Bernoulli model is slightly better than the other model at annotation.

4.3. Ranked Retrieval with Single Word Queries

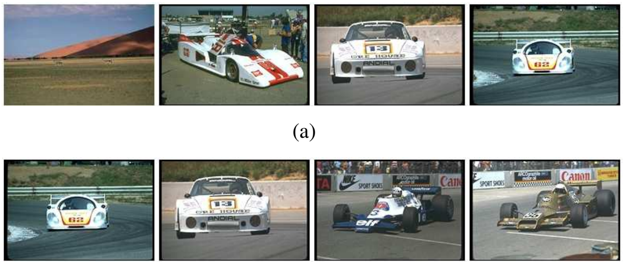
The annotation results reported above ignore rank order. That is, imagine that one wanted to find all car images. One

Model					
CRM-Rectangles	food outdoors monologue graphics_and_text text_overlay nonstudio_setting	face male_face indoors news_subject_monologue male_news_person	outdoors sky transportation water_body boat	nonstudio_setting people sport_event basketball face	graphics_and_text text_overlay monologue physical_violence gun_shot
MBRM	graphics_and_text text_overlay nonstudio_setting people_event face,male_face	face indoors studio_setting news_subject_monologue female_face female_news_person	outdoors sky transportation water_body boat	nonstudio_setting people sport_event basketball face	graphics_and_text text_overlay monologue physical_violence gun_shot

Figure 2: Top automatic annotations produced by the CRM-Rectangles and MBRM models. MBRM performs better than CRM-Rectangles for the first two images. For the last three images, the annotations are identical. Note that for many video frames are annotated with the words graphics_and_text and text_overlay because of the station logos - difficult to see in these images.

Models	CRM-Rectangles	MBRM
#words with recall ≥ 0	79	83
Results on all 110 words.		
Mean Per-word Recal	0.23	0.26
Mean Per-word Precision	0.23	0.26
Results on all words with recall ≥ 0		
Mean Per-word Recall	0.32	0.34
Mean Per-word Precision	0.32	0.35

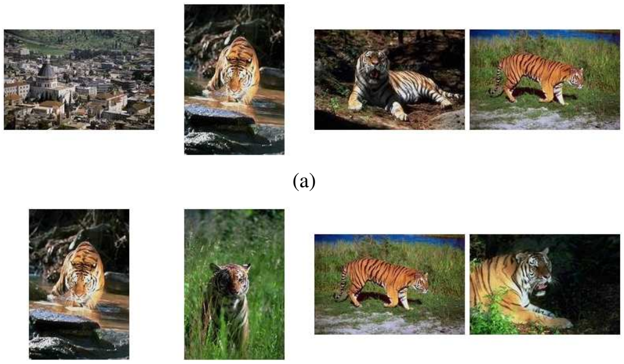
Table 2: Performance on the automatic annotation task on the Trec Video dataset. MBRM performs better than CRM-Rectangles.


(a)

(b)

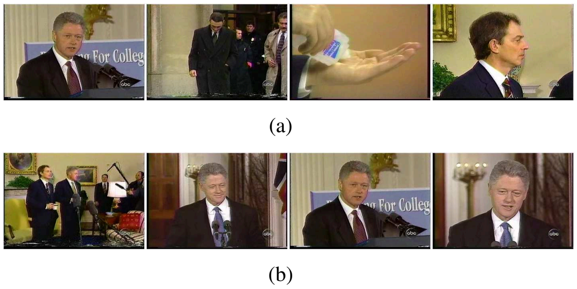
Figure 3: First 4 ranked results for the query "car" in the Corel collection using a) CRM-Rectangles and b) MBRM.

would ideally like to rank these according to the probability of annotation and hope that the top ranked ones are all cars. In fact, in large databases most users are not likely to even want to see more than 10 or 20 images in response to a query. Rank order is, therefore, very important for such applications. Figures 3-6 show the performance of CRM-Rectangles and MBRM in response to one word text queries. Although the annotation performance of the two models does not seem to be that different, the results show that the retrieval performance can be very different. To evaluate rank order, one can look at the performance on ranked retrieval in response to one word queries. Given a query word, the system will return all the images which are automatically annotated with that word, ranked according to the probabilities of that word generated by these images. We use a metric called mean average precision to evaluate the retrieval performances. Average precision is the average of precision values at the ranks where relevant (here 'relevant' means that the ground-truth annotation of this image contains the query word) items occurs, which is further averaged over all queries to give mean average precision. Table 3 shows that for ranked retrieval the Bernoulli model substantially outperforms (by 15% for the Corel dataset and by 16% for the Trec Video dataset) the multinomial model.


(a)

(b)

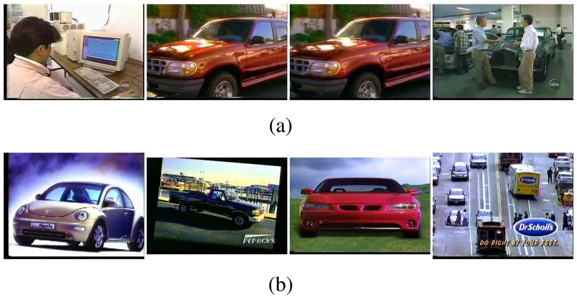
Figure 4: First 4 ranked results for the query "tiger" in the Corel collection using a) CRM-Rectangles and b) MBRM.



(a)

(b)

Figure 5: First 4 ranked results for the query “Bill_Clinton” in the Trec Video collection using a) CRM-Rectangles and b) MBRM. Note the first picture shows Bill Clinton with Toni Blair.



(a)

(b)

Figure 6: First 4 ranked results for the query “car” in the Trec Video collection using a) CRM-Rectangles and b) MBRM.

5. Summary and Conclusions

We have proposed a multiple-Bernoulli relevance model for image annotation, to formulate the process of a human annotating images. The results show that it outperforms, especially on the ranked retrieval task, the (multinomial) continuous relevance model and other models on both the Corel dataset and a more realistic Trec Video dataset. Future work will include a more extensive retrieval task with this model, which allows for longer text strings. Other extensions may include larger datasets, better features and more sophisticated models.

Mean Average Precision for Corel Dataset		
Models	All 260 words	Words with recall ≥ 0
CRM-Rectangles	0.26	0.30
MBRM	0.30	0.35
P-value	0.000037	0.0000082

Mean Average Precision for Video Dataset		
Models	All 110 words	words with recall ≥ 0
CRM-Rectangles	0.25	0.29
MBRM	0.29	0.37
P-value	0.000013	0.000000026

Table 3: Ranked retrieval results based on one word queries. MBRM performs much better than the multinomial model (CRM-Rectangles) - the rank ordering of the MBRM is much better than that produced by CRM-Rectangles. The P-value shows that the performance improvement is statistically significant according to the sign test.

References

[1] S. Agarwal and D. Roth. Learning a Sparse Representation for Object Detection, IN *Proc. ECCV*, pages 113-130, 2002.

[2] K. Barnard, P.Duygulu, N. de Freitas, D. Forsyth, D. Blei, and M.I.Jordan. Matching Words and Pictures. In *Journal of Machine Learning Research*, Vol 3, pp 1107-1135, 2003.

[3] D. Blei, and M. I. Jordan. (2003) Modeling annotated data. In *Proceedings of the 26th Intl. ACM SIGIR Conf.*, pages 127–134, 2003

[4] P. Carbonetto, N. de Freitas. Why can’t Jos read? The problem of learning semantic associations in a robot environment. In *Human Language Technology Conference Workshop on Learning Word Meaning from Non-Linguistic Data*, 2003.

[5] P. Duygulu, K. Barnard, N. de Freitas, and D. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In ECCV’02, pages 97-112, 2002.

[6] R. Fergus, P. Perona and A Zisserman Object Class Recognition by Unsupervised Scale-Invariant Learning. IN *Proc. CVPR’03*, vol II pages 264-271, 2003.

[7] J. Jeon, V. Lavrenko and R. Manmatha. (2003) Automatic Image Annotation and Retrieval using Cross-Media Relevance Models In *Proceedings of the 26th Intl. ACM SIGIR Conf.*, pages 119–126, 2003

[8] V. Lavrenko, R. Manmatha and J. Jeon. A Model for Learning the Semantics of Pictures, In the *Proceedings of NIPS’03 16, 2004*.

[9] H. Schneiderman, T. Kanade. A Statistical Method for 3D Object Detection Applied to Faces and Cars. *Proc. IEEE CVPR 2000*: 1746-1759

[10] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[11] K. Sung and T. Poggio. Example-based Learning for View-based Human Face Detection. In *IEEE PAMI*, 20(1):39-51, Jan 1998.

[12] http://www-nlpir.nist.gov/projects/trecvid

[13] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. CVPR’01*, pages 511-518, 2001.

[14] T. Westerveld and A.P. de Vries. Experimental Evaluation of a Generative Probabilistic Image Retrieval Model on ’Easy’ Data. In *Proceedings of the SIGIR Multimedia Information Retrieval Workshop 2003*, Aug, 2003.