Regularizing Translation Models for Better Automatic Image Annotation

Feng Kang
Department of Computer Science and Engineering
Michigan State University
East Lansing, MI 48824
kangfeng@msu.edu

Rong Jin
Department of Computer Science and Engineering
Michigan State University
East Lansing, MI 48824
rongjin@cse.msu.edu

Joyce Y. Chai
Department of Computer Science and Engineering
Michigan State University
East Lansing, MI 48824
jchai@cse.msu.edu

ABSTRACT
The goal of automatic image annotation is to automatically generate annotations for images to describe their content. In the past, statistical machine translation models have been successfully applied to automatic image annotation task [8]. It views the process of annotating images as a process of translating the content from a 'visual language' to textual words. One problem with the existing translation models is that common words are usually associated with too many different image regions. As a result, uncommon words have little chance to be used for annotating images. Uncommon words are important for automatic image annotation because they are often used in the queries. In this paper, we propose two modified translation models for automatic image annotation, namely the normalized translation model and the regularized translation model, that specifically address the problem of common annotated words. The basic idea is to raise the number of blobs that are associated with uncommon words. The normalized translation model realizes this by scaling translation probabilities of different words with different factors. The same goal is achieved in the regularized translation model through the introduction of a special Dirichlet prior. Empirical study with the Corel dataset has shown that both two modified translation models outperform the original translation model and several existing approaches for automatic image annotation substantially.

Categories and Subject Descriptors
H.3.3 [**Information Search and Retrieval**]: *Retrieval models*

General Terms
Algorithms, Measurement, Experimentation.

Keywords
Automatic image annotation, translation model, regularized translation model, normalized translation model.

CIKM'04, November 8-13, 2004, Washington, DC, USA.

1. INTRODUCTION
Efficient access to image database requires the ability to search and organize images effectively. Although images could be retrieved based on their features such as color, texture, it is usually more natural and desirable for users to search image databases using textual queries. Furthermore, textual queries are usually more accurate than image queries in terms of expressing the users' information needs. For example, consider that a user is looking for images of a tiger. If the query image that he uses is a photo of a tiger in grass, based on the match of image features, many of the retrieved images will be just pictures of grass without any tigers. This is because it is unclear to the system which object between the grass and the tiger is looked for by the user. On the other hand, a textual query such as 'Find me photos of tigers' can clearly convey the information need of the user.

The key to image retrieval using textual queries is image annotation. With annotated words for images, a problem of image retrieval becomes a problem of textual retrieval and many well-developed textual retrieval algorithms such as language modeling approaches [10, 11, 17, 19] can be applied to find images that are relevant to textual queries. Since manual annotation is usually expensive and subjective, many methods have been developed to annotate images automatically [1-3, 5, 6, 8, 9, 12-16].

Many of the automatic annotation methods applied machine-learning techniques to first learn the correlation between image features and textual words from the annotated training images and then apply the learned correlation to predict words for unseen images. A machine translation model for automatic image annotation [8] views the process of annotating images as a process of translating information from a 'visual language' to textual words. Images are first segmented into different regions, which are further grouped into a number of clusters, or image blobs as called in [8]. Then, correspondence between image blobs and annotated words is learned through a statistical machine translation model [4]. Finally, the learned model is applied to 'translate' un-annotated images into textual words. Compared to other models for automatic image annotation, such as classification approaches [5, 6, 13] and latent space models [3, 15], the machine translation model for automatic image annotation has the advantage in that words are annotated to image regions, not just the whole image. This is useful information for object recognition and can also be used to provide a better rank list of retrieved images. For example, if the textual query is 'Find me images of a tiger', it would be more desirable to rank the

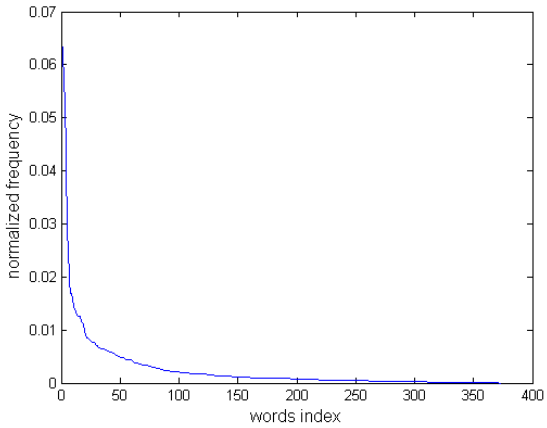

Figure 1. The distribution of term frequency for annotated words in a subset of Corel data**.**

retrieved images in the descending order based on the area of image regions that are annotated with the word 'tiger'.

One difficulty with translation models for automatic image annotation arises from the skewed distribution of word frequency. According to [4], one key for translation models to disambiguate the alignment between image regions and annotated words is co-occurrence statistics. If an image blob co-occurs more frequently with word '*A*' than the other words, it will be more likely for the image blob to be associated with '*A*'. According to [15], the term frequency of annotated words follows the Zipf's law, namely a small number of words appear very often in image annotations and most words are used only by a few images. Figure 1 plots the percentage of times that each word is used by image annotations in a subset of Corel dataset [8]. The problem with the co-occurrence statistics in automatic image annotation is further complicated by the noise in clustering the massive number of image regions into a small number of blobs. Because each image region is represented by a set of fixed features such as colors, textures and shapes, regions for different annotated words can have similar distributions over the space of image features and therefore are grouped into the same cluster, or the same image blob. As a result, a blob for a rare word can co-occur more frequently with a common word than the rare word. Using the previous example, consider that image regions for tulips are grouped together with image regions for other flowers. If most flowers are surrounded by grass, word 'grass' will co-occur more frequently with the blob for flowers than any single flower name. It is the inaccurate co-occurrence statistics that allow common annotated words to be associated with many irrelevant image blobs and thus degrade the quality of auto-annotations generated by the machine translation models.

In this paper, we propose two modified translation models, namely the normalized translation model and the regularized translation model that alleviate the above problem. The basic idea is to raise the number of blobs that are associated with uncommon words. The normalized translation model realized this by scaling translation probabilities of different words with different factors. The same goal is achieved in the regularized translation model through the introduction of a special Dirichlet prior. Empirical study with the Corel dataset has shown that both modified translation models outperform the original translation model and several existing approaches for automatic image annotation substantially.

The rest paper is organized as: section 2 summarizes the related work on automatic image annotation. In section 3, the regularized translation model is introduced and an efficient optimization algorithm is discussed. Experiment results are presented in section 4. Section 5 concludes this work.

2. RELATED WORK

In this section, we will first overview the previous work on automatic image annotation, followed by a detailed description of a machine translation model for automatic image annotation.

2.1 Overview of Methods for Automatic Image Annotation

A variety of machine learning methods have been applied to automatic image annotation, including machine translation model [8], co-occurrence model [16], latent space approaches [1, 15], graphic models [3], classification approaches [5, 6, 13], and relevance language models [9, 12]. The co-occurrence model [16] collects co-occurrence counts between words and image features and uses them to predict annotated words for images. Duygulu et al. [8] improved the co-occurrence model by utilizing machine translation models, in which the annotation procedure is analogous to the procedure of machine translation. Another way of capturing co-occurrence information is to introduce latent variables that link image features with words. Standard latent semantic analysis (LSA) and probabilistic latent semantic analysis (PLSA) are applied to automatic image annotation [15]. Barnard et al. [1] introduced a hierarchical aspect model for image annotation in order to account for the fact that some words are more general than others. More sophisticated graphical models, such as Gaussian Mixture Model (GMM), Latent Dirichlet Allocator (LDA), and correspondence LDA, have also applied to the image annotation problem recently [3]. The classification approaches for automatic image annotation treat each annotated word as an independent class and create a different image classification model for every word. Work such as linguistic indexing of pictures [13], image annotation using SVM [6] and Bayes point machine [5] fall into this category. More recently, relevance language models have been applied to automatic image annotation [9, 12]. It first finds images from the training set that are similar to the test image and then combine annotated words of similar training images together as the annotation for the test image. Empirical studies [9, 12] have shown that relevance language models for image annotation are better than translation models.

2.2 A Machine Translation Model for Automatic Image Annotation

In a translation model for automatic image annotation, images are segmented into multiple regions. Regions from different images are grouped together into a number of clusters, or image blobs [8], to form a visual vocabulary.

Let the collection of annotated images denoted by *T*, and the size of the collection denoted by $|T|$. Each annotated image $J_i \in T$ is represented by its image blobs and annotated words, i.e.,

$J_i = \{\vec{b}_i; \vec{w}_i\} = \{b_{i,1}, b_{i,2,}..., b_{i,m}; w_{i,1}, w_{i,2}, ..., w_{i,n}\}$. m and n are the number of blobs and words in image i respectively; $b_{i,j}$ is the number of j-th blob that appears in the i-th image; $w_{i,j}$ is a binary variable that indicates whether or not the j-th word appears in the i-th image.

Using the IBM model 1 of translation model [8], the probability of annotating image blobs $\vec{b}_i = \{b_{i,1}, b_{i,2}, ..., b_{i,m}\}$ with words $\vec{w} = \{w_{i,1}, w_{i,2}, ..., w_{i,n}\}$, i.e., $p(\vec{w}_i \mid \vec{b}_i)$, can be expressed as follows:

$$p(\vec{w}_i \mid \vec{b}_i) = \prod_{j=1}^{n} p(w_{i,j} \mid \vec{b}_i) \propto \prod_{\{j|w_{i,j}=1\}} \sum_{k=1}^{m} t_{j,k} b_{i,k} \qquad (1)$$

where $t_{j,k}$ stands for the probability of translating the k-th blob into the j-th word and is subject to the constraint $\sum_j t_{j,k} = 1$, namely each blob has to be translated into one of the annotated words. In order to annotate an image $I = \{b_1, b_2, ..., b_m\}$, Equation (1) is applied to find the set of words $\vec{w}$ that maximizes $p(\vec{w} \mid \vec{b})$.

The key to the translation model for image annotation is the set of translation probabilities $\{t_{j,k}\}$. These probabilities can be obtained by maximizing the likelihood of annotated training images, i.e.,

$$l(T) = \prod_{i=1}^{|T|} p(\vec{w}_i \mid \vec{b}_i) = \prod_{i=1}^{|T|} \prod_{\{j|w_{i,j}=1\}} \sum_{k=1}^{m} t_{j,k} b_{i,k} \qquad (2)$$

The Expectation-Maximization (EM) algorithm [7] is applied to find the optimal solution for Equation (2), which iteratively updates the translation probabilities using the following equation:

$$t_{j,k}^{new} = \frac{1}{Z_k} \sum_i \frac{w_{i,j} b_{i,k} t_{j,k}^{old}}{\sum_l b_{i,l} t_{j,l}^{old}} \qquad (3)$$

where $t_{j,k}^{old}$ and $t_{j,k}^{new}$ stand for the translation probability for the previous and current iteration, respectively. Z_k is a normalization factor that ensures $\sum_j t_{j,k}^{new} = 1$. According to Equation (3), a common word will have large translation probabilities for many different blobs since its term frequency $w_{i,j}$ is non-zero for a large number of training examples. To further illustrate this problem, based on the subset of Corel data [8], we computed the number of dominant blobs associated with each word. For a word j, dominant blob is defined as a blob, which has the dominative translation probabilities to this word than other words, or $\{k \mid t_{j,k} > t_{l,k} \ \forall l \neq j\}$. The results are plotted in Figure 2. According to Figure 2, common words are associated with many more blobs than uncommon words. Particularly, many rare words are associated with no blobs, which

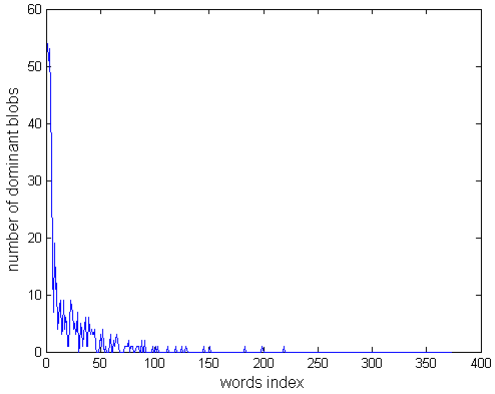

Figure 2. Plot of the number of blobs that are associated with each word. Horizontal axis are sorted in the descending order of their term frequency in image annotations.

makes it almost impossible to predict them given any visual content (i.e., image blobs).

In this paper, we only consider the IBM model 1 for statistical translation model, in which a uniform distribution is applied for alignment probabilities. Some papers consider IBM model 2 for automatic image annotation, which takes into account a non-uniform distribution for alignment probabilities. However, there have been issues with how to correctly define alignment probabilities within the context of automatic image annotation [1, 9]. Furthermore, our experience with IBM model 2 reveals a poorer performance compared to model 1. As a result, in this paper, we limited the discussion to the translation model 1.

3. MODIFIED TRANSLATION MODELS FOR IMAGE ANNOTATION

In this section, we first introduce an ad-hoc approach, named 'normalized translation model', that alleviates the aforementioned problem with the translation model 1. Then, a better solution, named 'regularized translation model', is introduced by adding a model prior to the translation model 1.

3.1 A Normalized Translation Model

In order to reduce the unbalance in the distribution of the number of blobs associated with every word, we could ensure that each word is associated with at least one single blob with high probability. One solution is to adjust the maximum translation probability $t_{j,k}$ for each word j. Let $k(j)$ be the index of the blob that corresponds to the maximum translation probability for the j-th word, or $t_{j,k(j)} > t_{j,l} \ \forall l \neq k(j)$. Then, if we can ensure that $t_{j,k(j)}$, the maximum translation probability for the j-th word, dominates over other words in terms of translating blob $k(j)$, the j-th word is guaranteed to be associated with the blob $k(j)$. If this is true for every word, then every word has chance to appear in the annotation set if a certain corresponding blobs appear in the image, while in the translation model 1, it's possible that some words are associated with any blobs with too low probability and have no chance to appear in the annotation set for

test images. To realize this, we can scale translation probabilities of different words using different factors. One ad-hoc implementation of this idea is described as follows:

1) Apply the translation model 1 to compute translation probabilities $t_{j,k}$ between different words and image blobs.
2) 'Normalize' each translation probability $t_{j,k}$ by dividing it by $t_{j,k(j)}$, i.e., the maximum translation probability for the same word *j*, or $\tilde{t}_{j,k} = t_{j,k} / t_{j,k(j)}$. By doing so, each word *j* has at least one 'normalized' translation probability $\tilde{t}_{j,k} = 1$, which guarantees that at least one blob is associated with the word with high probability.
3) Apply the 'normalized' translation probabilities to predict annotated words for test images using Equation (1).

For later reference, we call this approach 'normalized translation model for automatic image annotation', or **NTM**.

3.2 A Regularized Translation Model

In this subsection, we will first present the framework of the regularized translation model for automatic image annotation, followed by the description of an efficient EM algorithm for finding the optimal solution to the modified translation model.

3.2.1 Description of Framework for the Regularized Translation Model

In order to address the problem with machine translation model more systematically, we propose another modified translation model for automatic image annotation, called 'regularized translation model'. The basic idea is to impose our prior knowledge of desired translation model in the selection of translation models. If each blob represents a different type of objects, we would expect that almost equal number of blobs is associated with each word, or at least each word is associated with a certain number of blobs. In the framework of Bayesian Learning, this prior preference of translation models can be introduced into the machine translation model 1 through an appropriate prior.

In order to form such a prior, the first question is to find an appropriate measurement that indicates the number of blobs that are associated with each word. In this paper, we use the normalized sum of translation probabilities for each word, i.e., $\beta_j = \frac{\sum_k t_{j,k}}{m}$ where *m* is the total number of blobs. Since measurement β_j is proportional to the sum of translation probabilities for the j-th word, it does provide a good indication of how many blobs are associated with the word *j*. Meanwhile, β_j can also be interpreted as the probability for the j-th word to be associated with any blobs. Particularly, β_j satisfies the axioms for probability, namely 1) $0 \le \beta_j \le 1$ and 2) $\sum_j \beta_j = 1$. Because of the probability interpretation for β_j, we can introduce a prior distribution for β_j that will indirectly influence the results for translation probabilities $t_{j,k}$.

The second question for forming a prior is to choose an appropriate distribution for β_j. Since the desired translation model is to have almost equal number of blobs to be associated with each word, a Dirichlet priori can be used for $\vec{\beta}$,

$$\Pr(\vec{\beta}) \sim Dirichlet(\vec{\beta}, \alpha) \propto \prod_{j=1}^{n} \beta_j^{\alpha} \tag{4}$$

where $\alpha > 0$ is the hyper-parameter that determines the shape of the Dirichlet distribution. Note that the maximum point for the above Dirichlet distribution is when β_j is a constant. Furthermore, the larger the α is, the narrower the distribution will be.

By adding the above prior to the translation model 1, the posterior probability for the training images is then modified into the following form:

$$\begin{aligned} l_{reg}(T) &= \Pr(\vec{\beta}) \prod_{i=1}^{|T|} p(\vec{w}_i \mid \vec{b}_i) \\ &\propto \prod_{l=1}^{n} \left(\sum_{s=1}^{m} t_{l,s} \right)^{\alpha} \prod_{i=1}^{|T|} \prod_{\{j \mid w_{i,j}=1\}} \sum_{k=1}^{m} t_{j,k} b_{i,k} \end{aligned} \tag{5}$$

Similar to the translation model 1, the optimal translation probabilities are obtained by maximizing the objective function in Equation (5). Compared to the translation model 1 in Equation (2), the objective function in Equation (5) requires that not only should the optimal translation probabilities explain well the correspondence between image blobs and annotated words but also be consistent with the prior preference on translation models, namely different words are associated with similar number of image blobs. Therefore, the resulting translation model from Equation (5) will be more desirable than the model obtained from Equation (2). For late reference, we call this modified translation model 'regularized translation model', or **RTM**.

3.2.2 An EM Algorithm for the Regularized Translation Model

With the regularized translation model in Equation (5), the next important question is how to efficiently obtain the optimal translation probabilities $t_{j,k}$ that maximize the function in Equation (5). The difficulty with optimizing Equation (5) lies in two aspects:

1) *It has a large number of parameters*. The number of parameters in translation models is $m \times n$, i.e., the number of blobs times the number of unique words. For the experiment conducted in this study, the number of parameters is close to 20,000.
2) *It is a constrained optimization problem*. The optimal solution to Equation (5) should satisfy the axioms of probability, namely $0 \le t_{j,k} \le 1 \ \forall j,k$ and $\sum_j t_{j,k} = 1 \ \forall k$

The above two aspects make it difficult to apply most traditional optimization approaches such as sequential quadratic programming [18] to the problem in Equation (5). In this paper, we present an EM algorithm for efficiently optimizing the objective function in Equation (5).

First, instead of optimizing the likelihood of training data in Equation (5), we can optimize the log-likelihood of training data, i.e.,

$$\Phi = \log\left(l_{reg}(T)\right) = \alpha\sum_{l=1}^{n}\log\left(\sum_{s=1}^{m} t_{l,s}\right) + \sum_{i=1}^{|T|}\sum_{j=1}^{n} w_{i,j}\log\left(\sum_{k=1}^{m} t_{j,k}b_{i,k}\right) \quad (6)$$

Then, following the idea of EM algorithm, we update the optimal solution iteratively. Particularly, at each iteration, we need to find a set of translation probabilities $\{t_{j,k}^{new}\}$ better than the old ones $\{t_{j,k}^{old}\}$ that are computed for the previous iteration. To this end, we can examine the difference in the log-likelihood between two consecutive iterations, i.e.,

$$\Phi - \Phi' = \alpha\sum_{l=1}^{n}\log\left(\frac{\sum_{s=1}^{m} t_{l,s}^{new}}{\sum_{s=1}^{m} t_{l,s}^{old}}\right) + \sum_{i=1}^{|T|}\sum_{j=1}^{n} w_{i,j}\log\left(\frac{\sum_{k=1}^{m} t_{j,k}^{new}b_{i,k}}{\sum_{k=1}^{m} t_{j,k}^{old}b_{i,k}}\right)$$
$$\geq \alpha\sum_{l=1}^{n}\sum_{s=1}^{m}\frac{t_{l,s}^{old}}{\sum_{s=1}^{m} t_{l,s}^{old}}\log\left(\frac{t_{l,s}^{new}}{t_{l,s}^{old}}\right) + \sum_{i=1}^{|T|}\sum_{j=1}^{n}\frac{w_{i,j}t_{j,k}^{old}b_{i,k}}{\sum_{k=1}^{m} t_{j,k}^{old}b_{i,k}}\log\left(\frac{t_{j,k}^{new}}{t_{j,k}^{old}}\right) \quad (6)$$

The new translation probabilities $\{t_{j,k}^{new}\}$ are obtained by maximizing the above difference, i.e.,

$$t_{j,k}^{new} = \frac{1}{Z_k}\left(\alpha\frac{t_{j,k}^{old}}{\sum_{l} t_{j,l}^{old}} + \sum_{i}\frac{w_{i,j}b_{i,k}t_{j,k}^{old}}{\sum_{l} b_{i,l}t_{j,l}^{old}}\right) \quad (7)$$

where Z_k is a normalization factor that ensures $\sum_j t_{j,k}^{new} = 1$.

Comparing the above updating equation to Equation (3), we can see that Equation (7) has an extra term $\alpha t_{j,k}^{old} / \left(\sum_{l} t_{j,l}^{old}\right)$ in the right hand side of the equation. For each word *j*, this extra term gives a greater share of α to $t_{j,k(j)}$, the maximum translation probability for the j-th word, than any other translation probabilities $t_{j,l}$ for the same word. As a result, the maximum translation probability for each word will benefit most from this extra term. This is consistent with the idea deployed in the 'normalized translation model' that has been discussed in Section 3.1. Furthermore, the updating equation in (7) is able to adjust the sum of translation probabilities for different words (i.e., $\sum_k t_{j,k}$) to be close. This is because according to Equation (7), a word *j* with a small sum of translation probabilities will get more promotion from term $\alpha t_{j,k}^{old} / \left(\sum_{l} t_{j,l}^{old}\right)$ than a word that has a large sum of translation probabilities. Note the difference between the promotion of translation probability from different blobs to the same words and the translation probability between words with different frequencies.

Comparison to the normal usage of Dirichlet priors. Note that the Dirichlet prior introduced in this work is different from the Dirichlet priors used by many other studies [3]. For most previous studies of Bayesian learning, Dirichlet priors simply introduce *constant* pseudo counts into the estimation of probabilities. However, here the pseudo count introduced by the Dirichlet prior (i.e., $\alpha t_{j,k}^{old} / \left(\sum_{l} t_{j,l}^{old}\right)$) is no longer a constant. In fact, it is this non-constant pseudo count that leads to a more balanced distribution in the number of blobs associated with each word.

The global optimum for the EM algorithm. Interestingly, the objective function in Equation (5) is strictly convex. Therefore, it does not have any local optimum and the EM algorithm presented in Equation (7) will guarantee to find the global optimal solution. This result can be easily understood by treating each term $\left(\sum_{s=1}^{m} t_{l,s}\right)^{\alpha}$ in the prior as α number of pseudo-annotated images that include all blobs in its picture and are annotated only by l-th word. As a result, the regularized model is almost identical to translation model 1 except that the regularized model uses both the pseudo-annotated images and the annotated images from the training dataset. Since the translation model for any number of annotated images is strictly convex [4], the new objective function in Equation (5) will be strictly convex.

The choice of α. As already revealed by the previous discussion, constant α has a great impact on the resulting translation model. A larger value for α will introduce more pseudo-annotated images and therefore result in a more balanced distribution for the number of blobs that is associated with each word. In the section for experiment, we provide a detailed study of how the value of α will influence the quality of auto-annotations.

4. EXPERIENTS

In this experiment, we will address the following four questions for the proposed translation models:

1) *How effective are the proposed translation models compared to the translation model 1 for automatic image annotation*? In this experiment, we compared both the 'normalized translation model' and the 'regularized translation model' to the translation model 1.

2) *Which modified translation model is more effective*? In this experiment, we compared the 'regularized translation model' to the 'normalized translation model' for automatic image annotation.

3) *How does the constant α influence the quality of the resulting translation models*? In this experiment, we varied the value of α over a large range to see its impact on the quality of resulting translation models.

4) *How effective are the modified translation models compared to other annotation models*? In this experiment, we compared the modified translation models to the relevance language model for automatic image annotation which has shown good performance in the recent studies [9, 12].

	Translation Model 1	Normalized Translation Model	Regularized Translation Model	Relevance Language Model
#Ret_Query	62	89	90	76
Ave. recall	0.1988	0.2501	0.2921	0.2513
Ave. precision	0.1688	0.2204	0.2181	0.2176

Table 1. Results for the translation model 1, the normalized translation model (titled by NTM), the regularized translation model (titled by RTM), and the relevance language model. Union of predicted results has 112 words

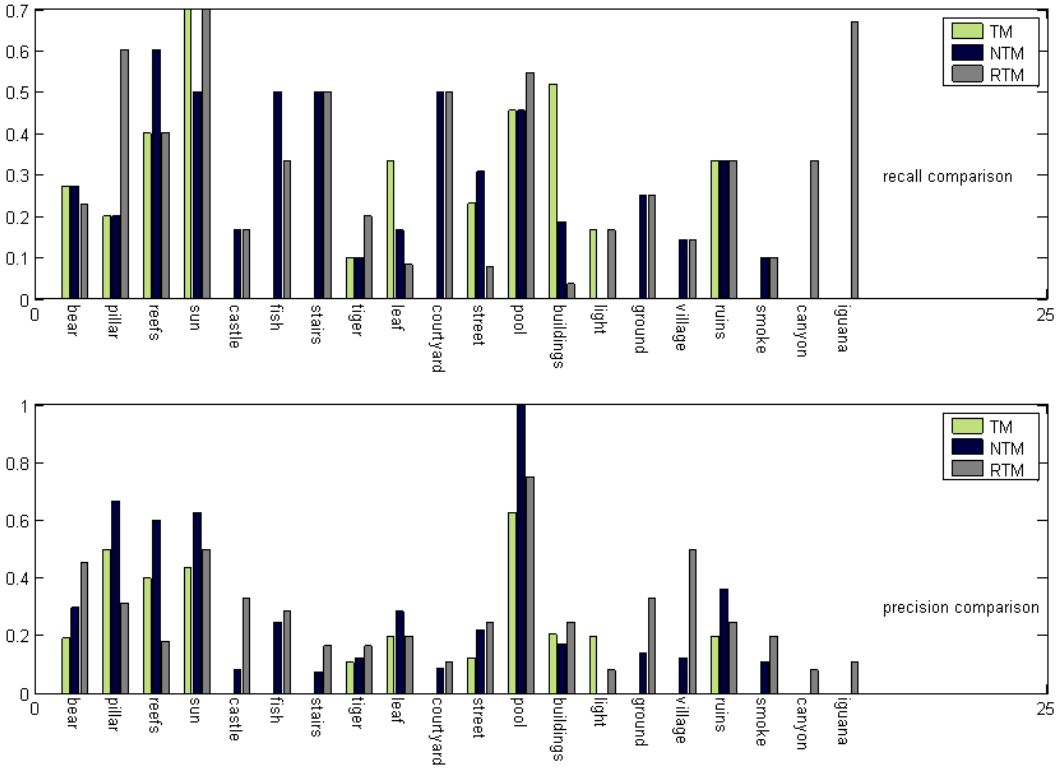

Figure 3. Average recall and precision of different translations models for 20 randomly selected words

4.1 Experiment Data

The same subset of Corel data used in [8] is used in this experiment. It consists of 5000 annotated images with 4500 of them are used for training and the remaining 500 images used for testing. Each image is segmented into multiple regions that are represented by 33 different image features, such as colors, textures and shapes [8]. Regions from different images are clustered into 500 image blobs using the K-means algorithm. 371 different words are used for annotating both training and testing images.

Similar to the previous studies on automatic image annotation, the quality of automatic image annotation is measured by the performance of retrieving auto-annotated images regarding to single-word queries. For each single-word query, **precision** and **recall** are computed using the retrieved lists that are based on the true annotations and the auto-annotations. Let I_j be a test image, t_j be its true annotation, and g_j be its auto-annotation. For a given query word *w*, precision and recall are defined respectively as:

$$\text{precision}(w) = \frac{|\{I_j \mid w \in t_j \wedge w \in g_j\}|}{|\{I_j \mid w \in g_j\}|}$$

$$\text{recall}(w) = \frac{|\{I_j \mid w \in t_j \wedge w \in g_j\}|}{|\{I_j \mid w \in t_j\}|}$$

The precision(*w*) measures the accuracy in annotating images with word *w* and the recall(w) measures the completeness in annotating images with word *w*. The average precision and recall over different single-word queries are used to measure the overall quality of automatically generated annotations for images. The third metric is the number of single-word queries for which at

Images	TM	NTM	RTM	Manual
	sky water tree forest cat	water boats forest bengal angelfish	forest cat tiger bengal canal	forest cat tiger bengal
	sky water people sand street	sand wall skyline street hotel	skyline hotel iguana marine formula	rocks iguana lizard marine
	water tree people grass buildings	tree grass grizzly fox man	anemone horses albatross foals man	field horses mare foals
	water people field coral ocean	city people anemone pack locomotive	anemone path pots canyon pack	leaf pots
	tree people buildings stone temple	wall roofs stone statue temple	stone temple pillar mosque sphinx	stone temple sculpture pillar

Table 2: Examples of annotations generated by the translation model (titled by TM), the normalized translation model (titled by NTM), and the regularized translation model (titled by RTM). The manual annotations are included in the last column.

least one relevant image can be retrieved using the auto-annotations, or **#Ret_Query**. It is defined as:

$$\#\text{Ret\_Query} = |\{w \mid \text{precision}(w) > 0 \wedge \text{recall}(w) > 0\}|$$

Note that this metric compensates the metrics of average precision and average recall by providing information about how wide is the range of words that contribute to the average precision and recall. This metric is important because a biased model can achieve high precision and recall value by only performing extremely well on a small number of queries with common words.

4.2 Experiment I: Modified Translation Models vs. the Translation Model 1

The results for both the translation model 1 and the two modified translation models are presented in Table 1. The average precision and recall are computed over the union set of single word queries that have at least one relevant image retrieved by any of four annotation methods. According to Table 1, both modified translation models are able to achieve substantially better performance than the translation model 1 in all three metrics. The most noticeable improvement is on the average recall and #Ret_Query: the average recall is increased from 20% to 25% and 29% for the two modified translation models respectively, and #Ret_Query is increased from 60 words to around 90 words. This improvement can be understood by the fact that both modified translation models tend to improve their ability on predicting uncommon words for images. As a result, more queries with single uncommon words are returned with relevant images, which leads to the improvement in both #Ret_Query and average recall. It is important to notice that the recall is improved without sacrificing the precision, which usually happens in textual

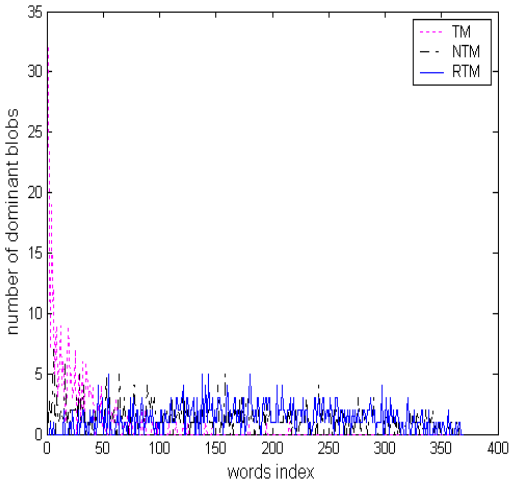

Figure 4. Distributions of the number of dominant blobs associated with each word for the original translation model, the normalized translation model (i.e., NTM), and the regularized translation model (i.e., RTM).

retrieval.

In addition to the average precision and recall, we also study the behavior of precision and recall for individual words. In Figure 3, we plot the precision and recall of 20 randomly selected words for the three different annotation models: the lightest bars correspond to the results for the translation model 1, the medium dark bars correspond to the results for the regularized translation model, and the darkest bars correspond to the results for the normalized translation model. According to Figure 3, for most words, the precisions and recalls achieved by the two modified translation models are better than the results for the translation model 1. For words such as 'arctic' and 'ground', the translation model 1 is unable to find any relevant images while the two modified models achieve reasonably decent precisions and recalls.

Furthermore, in Table 2, we include the annotations of five images that are generated by the three different translation models. For the purpose of comparison, we also include the manual annotations in the last column of Table 2. Notice that for all four images, the translation model 1 always puts the common word 'water' into the auto-annotations even though none of the four images contains water. In contrast, the two modified translation models do not have this problem. Only in the first example did the normalized translation model generate word 'water' for the annotation.

Finally, in order to illustrate the ability of the proposed models on adjusting the unbalanced distribution in the number of blobs associated with words, we computed the distributions for the two modified translation models. Figure 4 shows the two distributions. Compared with the number of blob for translation model 1 plotted in figure 2, the distributions for the two modified translation models are much more flat than the translation model 1. The impact is mainly on the head part and the tail part of the distribution. For the most common words (i.e., the head of the distribution), the number of associated blobs has been reduced dramatically. Meanwhile, the number of associated blobs has been raised substantially for the rare words (i.e., the tail part of the distribution). Particularly, words with zero number of associated blobs have diminished for these two modified translation models.

4.3 Experiment II: Regularized Translation Model vs. Normalized Translation Model

The results of average precision, average recall and #Ret_Query for the normalized translation model and the regularized translation model have already been listed in Table 2. Both models achieve similar performance in terms of the average precision and #Ret_Query (21% for the average precision and around 90 words for #Ret_Query). However, the regularized translation model outperforms the normalized translation model substantially in the average recall, with 28% versus 24%.

Although the goal of both modified translation models is to raise the number of blobs that are associated with rare words, different strategies are used. In the normalized translation model, this goal is achieved by scaling the translation probabilities of different word with different factors. The consequence of this operation is that the sum of translation probabilities for each blob is no longer a constant. In fact, the sum of translation probabilities can be much larger for some blobs than the others, which means that

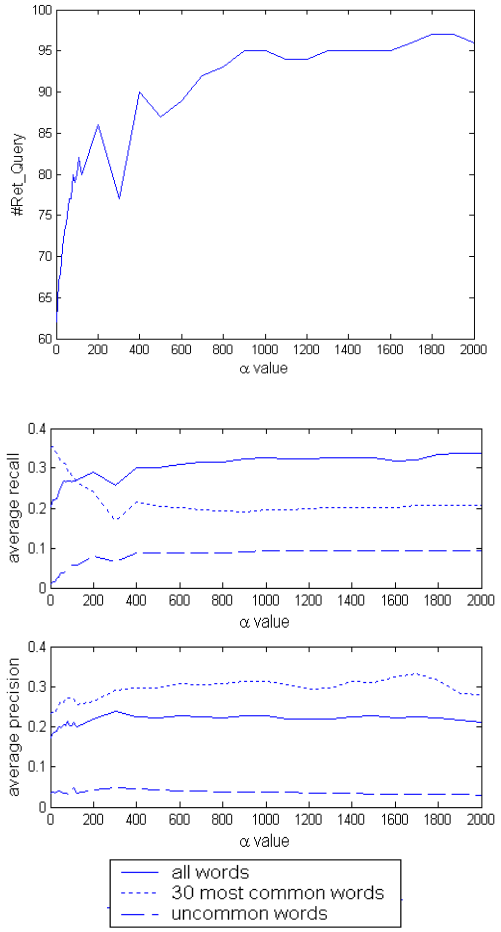

Figure 5: Change of #Ret_Query, average precision, and average recall for different value of α

more words are associated with some blobs than the others. In other words, the normalized translation probabilities lead to an unbalanced distribution for the number of words that are associated with each blob. In the regularized translation model, this unbalance is removed by the iterative application of E-step and M-step. It is this property that makes the regularized translation model better in retrieving images for single word queries and thus increases its overall average recall.

4.4 Experiment III: Impact of Constant α on the Regularized Translation Model

As already pointed out in the previous sections, constant α has a large impact on the performance of regularized translation model for automatic image annotation. In this experiment, we measure the change in the three metrics with regarding to different values for α. Figure 5 plots the curve for #Ret_Query, average precision, and average recall when constant α is varied from 0 to 2000.

It is interesting to observe that all the three metrics increase almost monotonically with α when α is less than 1500. When α is larger than 1500, the average precision begins to drop although the average recall still increases. Over all, the three

metrics have been improved substantially when α is increased from 0 to 200. However, the improvements become marginal after 200 and begin to saturate when α is larger than 400. This is because, when α is a small value, the distribution of the number of blobs associated with words is rather skewed and thus increasing the value for α will have a great impact on balancing the distribution, which leads to significant improvement for the three metrics. However, when α is the large value, the number of blobs associated words has already been evenly distributed over different words. As a result, increasing value for α will make almost no adjustment of the distribution of blob numbers and therefore little change will be made to the translation model.

Another surprising observation from Figure 5 is that the average recall is increased without the sacrifice of the average precision. In fact, both average precision and recall are increased substantially when α is increased from 0 to 200. This is contradictory to many studies in information retrieval, in which improvement in recall usually leads to degradation in precision. To have a better understanding of this phenomenon, we divide the words into two groups: a group of common words and a group of uncommon words. In Figure 5, in addition to the average precision and recall, we also plot the curve for the average precision and recall for both common words and uncommon words using the dotted lines and dashed lines, respectively. According to Figure 5, for both common words and uncommon words, the change in precision and recall follows the normal patterns, namely increase in recall is usually accompanied with decrease in precision. Furthermore, the trends in the change of precision and recall for these two groups of words are almost opposite to each other: increase in the recall of uncommon words is usually accompanied with a decrease in the recall of common words. Since the overall average value for precision and recall is the mean of precision and recall for these two groups of words, the opposite trends in these two groups somehow compensate each other and lead to increase in both the average precision and recall.

The fact that a larger α always results in overall better performance makes the regularized translation model a desirable algorithm for automatic image annotation. One disadvantage of using large values for α is that a larger α usually results in a slower convergence for the EM algorithm. Apparently, the performance of the regularized model saturates after α is set to be 400, α = 400 has the best tradeoff between computational cost and predication accuracy.

4.5 Experiment IV: Comparison to Other Annotation Models

In this subsection, we compared the modified translation models to the cross-media relevance model for automatic image annotation that has shown a substantially better performance than the translation model [8] in the recent studies [9, 12]. Since our model use the blobs to represent image regions, in this experiment, we will only compare to the blob-based cross-media relevance model [9] not the one using continuous feature values[12].

The results for both the modified translation models and the relevance annotation models are listed in Table 1. The three models achieve similar performance in the average precision with 21% accuracy. The two modified translation models achieve substantially better #Ret_Query than the relevance language model, with 90 words for both modified translation models and only 76 words for the relevance language model. Finally, the normalized translation model achieves similar performance in the average recall as the relevance language model while the average recall for the regularized model is substantially better than the relevance language model. Base on the above discussion, we conclude that the regularized translation model is superior to the relevance language model for automatic image annotation.

5. CONCLUSION AND FUTURE WORK

In this paper, we examined the problem with applying the existing translation model to automatic image annotation. Due to the skewed distribution of term frequency for annotated words, a few words are used much more frequently than most other words. As a result, common words are associated with many more image blobs than other uncommon words and is much more likely to be predicted for annotation than necessary. Two modified translation models are proposed to address this problem. Both of them try to balance the distribution of the number of blobs associated with words, particularly for the rare words. In the normalized translation model, translation probabilities are scaled for each word such that the maximum probability for the word is set to be 1. In the regularized translation model, a model prior is introduced into the translation model 1 to favor the models that the number of blobs associated each words is evenly distributed. Empirical studies with Core dataset have shown that the two modified translation models are able to improve the performance of the translation model substantially in terms of precision, recall and #Ret_Query (i.e., the number of words that are being predicted correctly at least once). Comparison to the relevance language model for automatic image annotation also indicated that the two modified translation models are superior. Finally, the empirical studies also revealed that the regularized translation model performs better than the normalized translation model, particularly in terms of recall.

Though this work is limited to the blob representation for image regions, we plan to apply the modified translation models to the feature representation of image region. This is because clustering methods usually introduce noise into the representation for image regions. For example, it can group two irrelevant image regions into the same cluster. Another interesting dimension that can be explored under the framework of translation model is to take advantage of the word correlation. Currently, the IBM model 1 treats the annotated words as independent entities and the correlation between different annotated words are simply ignored. A better translation model for automatic image annotation should take in account the word correlation for better predicting annotated words for images.

6. REFERENCES

[1] Barnard, K., P. Duygulu, and D. Forsyth. *Clustering Art*. in *Proceedings of the 2001 IEEE Computer Society Conference on Pattern Recognition*. 2001.

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

[3] Blei, D. and M. Jordan. *Modeling annotated data*. in *Proceedings of 26th International Conference on Research and Development in Information Retrieval (SIGIR)*. 2003.
[4] Brown, P., S. D. Pietra, V. D. Pietra, and R. Mercer, *The Mathematics of Statistical Machine Translation.* Computational Linguistics, 1993. **19**(2): p. 263-311.
[5] Chang, E., K. Goh, G. Sychay, and G. Wu, *CBSA: content-based soft annotation for multimodal image retrieval using bayes point machines.* CirSysVideo, 2003. **13**(1): p. 26-38.
[6] Cusano, C., G. Ciocca, and R. Schettini. *Image annotation using SVM*. in *Proceedings of Internet imaging IV, Vol. SPIE 5304*. 2004.
[7] Dempster, A. P., N. M. Laird, and D.B. Rubin, *Maximum likelihood from incomplete data via the EM algorithm.* Joural of Royal Statistical Society, 1977. **39**(1): p. 1-38.
[8] Duygulu, P., K. Barnard, N. d. Freitas, and D. A. Forsyth. *Object recognition as machine translation: learning a lexicon for a fixed image vocabulary*. in *Proceedings of 7th European Conference on Computer Vision*. 2002.
[9] Jeon, J., V. Lavrenko, and R. Manmatha. *Automatic Image Annotation and Retrieval using Cross-Media Relevance Models*. in *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval*. 2003.
[10] Jin, R., C. X. Zhai, and A. G. Hauptmann. *Title Language Model for Information Retrieval*. in *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2002)*. 2002.
[11] Lavrenko, V. and B. Croft. *Relevance-based langauge models.* in *The 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2001.
[12] Lavrenko, V., R. Manmatha, and J. Jeon. *A Model for Learning the Semantics of Pictures*. in *Proceedings of Advance in Neutral Information Processing*. 2003.
[13] Li, J. and J. Z. Wang, *Automatic linguistic indexing of pictures by a statistical modeling approach,*. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2003. **25**(19): p. 1075-1088.
[14] Maron, O., *Learning from Ambiguity*. 1998, MIT.
[15] Monay, F. and D. Gatica-Perez. *On Image Auto-Annotation with Latent Space Models.* in *Proc. ACM International Conference on Multimedia*. 2003.
[16] Mori, Y., H. TAKAHASHI, and R. Oka. *Image-to-Word Transformation Based on Dividing and Vector Quantizing Images With Words*. in *MISRM'99 First International Workshop on Multimedia Intelligent Storage and Retrieval Management*. 1999.
[17] Ponte, J., *A Language Modeling Approach to Information Retrieval*, in *Department of Computer Science*. 1998, Univ. of Massachusetts at Amherst.
[18] Schittowski, K., *NLQPL: A FORTRAN-Subroutine Solving Constrained Nonlinear Programming Problems.* Annals of Operations Research, 1985. **5**: p. 485-500.
[19] Zhai, C. X. and J. Lafferty. *Model-based feedback in the KL-divergence retrieval model*. in *Proceedings of the Tenth International Conference on Information and Knowledge Manage-ment (CIKM 2001)*. 2001.