Information-Theoretic Semantic Multimedia Indexing

João Magalhães[1], Stefan Rüger[1,2]

[1]Department of Computing
Imperial College London
South Kensington Campus
London SW7 2AZ, UK

[2]Knowledge Media Institute
The Open University
Walton Hall
Milton Keynes MK7 6AA, UK

(j.magalhaes@imperial.ac.uk, s.rueger@open.ac.uk)

ABSTRACT

To solve the problem of indexing collections with diverse text documents, image documents, or documents with both text and images, one needs to develop a model that supports heterogeneous types of documents. In this paper, we show how information theory supplies us with the tools necessary to develop a unique model for text, image, and text/image retrieval. In our approach, for each possible query keyword we estimate a maximum entropy model based on exclusively continuous features that were pre-processed. The unique continuous feature-space of text and visual data is constructed by using a minimum description length criterion to find the optimal feature-space representation (optimal from an information theory point of view). We evaluate our approach in three experiments: only text retrieval, only image retrieval, and text combined with image retrieval.

Categories and Subject Descriptors

H.3.1 [**Content Analysis and Indexing**]: Abstracting methods.

General Terms

Algorithms, Measurement, Experimentation.

Keywords

Multimedia indexing, minimum description length, multi-modal categorization, information retrieval.

1. INTRODUCTION

Demand for techniques that handle both text and image based documents is increasing with the wide spread of search applications. It is impossible to conceive nowadays a world without systems that allow us to search for specific news articles, scientific papers, or information in general. Users want more: they want to submit the same query to search for text documents, visual documents, or documents with both media, e.g., photographs with captions, video shots (keyframes and speech). To achieve this, a new breed of information retrieval models is required: one that seamlessly integrates heterogeneous data. We propose an information retrieval model that can simultaneously model text-only documents, image-only documents, and documents with both text and images.

1.1 Information Theory and Multimodal Data

The objective of the proposed information retrieval model is to create a single text and image model of each keyword from a given query vocabulary. Prior to estimating a text and image model for each keyword, we must first process the input text and image data, after which we can estimate the model of a given query keyword.

Both types of data have very different characteristics: while text data is typically sparse and high-dimensional, visual data is usually dense and low-dimensional (note that adjectives *high* and *low* are used to contrast the different data that we are dealing with). Information theory [9] provides us with a set of information measures that not only assess the amount of information that one single source of data contains, but also the amount of information that two sources of data have in common. After selecting the *optimal* text feature space and the *optimal* image feature-space with the minimum description length (MDL) criterion we merge both feature spaces and obtain a unique continuous feature space for text and visual data. Note we use "*optimal*" in this paper from an information theory point of view.

Once we have the optimal continuous feature space we estimate a maximum entropy model for each keyword present on the query vocabulary. To avoid over-fitting, a Gaussian prior on the parameters prevents situations where one single feature inserts bias in the model.

1.2 Contributions

In this paper we propose a maximum entropy model for each query keyword based on text and image features that were optimally pre-processed (from an information theory point of view). In contrast with previous maximum entropy contributions that work with only discrete and/or positive valued features, we use real-valued features allowing richer features to be included in our framework.

In our view, the first most important contribution in our approach is the ability to seamlessly integrate heterogeneous data (text only documents, image only documents, and documents with both text and images) in a unique information retrieval model that borrows strong statistical foundations from information theory.

The second most important contribution is in terms of information

CIVR'07, July 9-11, 2007, Amsterdam, The Netherlands.

retrieval system scalability. As will be discussed later, the framework has a very low computational cost for indexing and searching, and it can easily scale with the number of keywords in the vocabulary. Last but not least, the retrieval precision of the algorithm is in the same range as other state-of-the art algorithms.

This paper is organised as follows: Section 2 contrasts our work with previous work. In Section 3 we describe how text is handled as sparse data, and images as dense data; Section 4 describes the maximum entropy model implementation. Section 5 presents experiments and results. We conclude by discussing the proposed information-theoretic framework characteristics.

2. RELATED WORK

In text retrieval the search process is triggered by a text query that can be compared directly to the corpus of the documents in the collection. Since we want to offer a common query interface for both text and images we need to define a common vocabulary of keywords to query all possible types of documents. Therefore the present work is related to text categorization, image annotation and multimodal content annotation. We will now look at these three areas with a view to seamlessly integrate text and image data on the same framework.

Text categorization models pre-process data by removing stop-words and rare words, stemming, and finally term-weighting. Due to the high-dimensional feature space of text data most text categorization algorithms are linear models such as naïve Bayes [26], maximum entropy [28], Support Vector Machines [19], regularized linear models [44], and Linear Least Squares Fit [40]. Joachims [19] applies SVMs directly to the text terms. Text is ideal for applying SVMs without the need of a kernel function because data is already sparse and high-dimensional. Linear models fitted by least squares such as the one by Yang et. al [40] offer good precision, and in particular regularized linear methods, such as the one we propose and the one by Zhang and Oles [44], perform similarly to SVMs, with the advantage of yielding a probability density model. The maximum entropy classification model proposed by Nigam [28] defines a set of features that are dependent on the class being evaluated while we use a unique set of features for all keywords. The proposed maximum entropy framework has the same characteristics and performance of linear models (logistic regression, least squares) with the crucial advantage that while these approaches have no automatic mechanism to select a vocabulary size we use the minimum description length principle to select its optimal size.

Yang [39], and Yang and Liu [41] have compared a number of text classification algorithms and reported their performances on different text collections. Their results indicate that k-Nearest Neighbour, SVMs, and LLSF are the best classifiers. Note that nearest neighbour approaches have certain characteristics (see [14]) that make them computationally too complex to handle large-scale indexing.

The simplest image annotation models deploy a traditional multi-class supervised learning model and learn the class-conditional probability density distribution of each keyword w given its training data x. Bayes law is used to model $p(x \mid w)$, the features data density distribution of a given keyword. Several techniques to model $p(x \mid w)$ with different types of probability density distributions have been proposed: Yavlinsky et al. [43] deployed a nonparametric distribution; Carneiro and Vasconcelos [7] a semi-parametric density estimation; Westerveld and de Vries [37] a finite-mixture of Gaussians; while Vailaya et al. [36] apply a vector quantization technique. Density based approaches are among the most successful ones. However, density distributions are not adequate for text because the density models do not get enough support from such sparse data.

Other types of approaches are based on a translation model between keywords and images (global, tiles or regions). Inspired by automatic text translation research, Duygulu et al. [10] developed a method of annotating images with words. First, regions are created using a segmentation algorithm like normalised cuts. For each region, features are computed and then blobs are generated by clustering the image features for these regions across an image collection. The problem is then formulated as learning the correspondence between the discrete vocabulary of blobs and the image keywords. Following the same translation approach [11, 17, 20] have developed a series of translation models that use different models for keywords (multinomial/binomial) and images representations (hard clustered regions, soft clustered regions, tiles).

Hierarchical models have also been used in image annotation such as Barnard and Forsyth's [3] generative hierarchical aspect model inspired by a hierarchical clustering/aspect model. The data are assumed to be generated by a fixed hierarchy of nodes with the leaves of the hierarchy corresponding to soft clusters. Blei and Jordan [6] propose the correspondence latent Dirichlet allocation model; a Bayesian model for capturing the relations between regions, words and latent variables. The exploitation of hierarchical structures (either of the data or of the parameters) increases the number of parameters (model complexity) to be estimated with the same amount of training data.

Maximum entropy models have also been applied to image annotation [2, 18] and object recognition [21]. All these three approaches have specific features for each class (keywords in our case) which increases the complexity of the system. It is curious to note the large difference in precision results between [18] and [2], we believe that it is related to the lack of regularization and to a differing number of features. These approaches were not as successful as density estimation based models as maximum entropy works best in a high-dimensional sparse feature spaces. The proposed maximum entropy framework tackles this problem by expanding the feature space in a similar spirit to Hoffman's probabilistic Latent Semantic Indexing [15].

These single-modality based approaches are far from our initial goal but by analysing them we can see which family of models can be used to simultaneously model text, image, and multi-modal content. Each modality captures different aspects of that same reality, thus carrying valuable information about each keyword of the vocabulary. The simplest approach to multi-modal analysis is to design a classifier per modality and combine the output of these classifiers. Westerveld, et al. [38] combine the visual model and the text model under the assumption that they are independent, thus the probabilities are simply multiplied. Naphade and Huang [27] model visual features with Gaussian Mixtures Models (GMM), audio features with Hidden Markov Models (HMM) and combine them in a Bayesian network.

In multimedia documents the different modalities contain co-occurring patterns that are synchronised/related in a given way because they represent the same reality. Synchronization/relation and the strategy to combine the multi-modal patterns is a key point of the Semantic pathfinder system proposed by Snoek et al. [34, 35]. Their system uses a unique feature vector that concatenates a rich set of visual features, text features from different sources (ASR, OCR), and audio features. Three types of classifiers are available: logistic regression (which without regularization is known to over-fit [8]), Fisher linear discriminant, and SVMs (offering the best accuracy). The fusion of the different modalities is possible to be done at different levels and it is chosen by cross-validation for each concept. The extremely high computational complexity required to compute the visual features and to iteratively select the best classifier, the best type of fusion, and the SVMs parameter optimization are serious drawbacks of this system. IBM's Marvel system [1] has a similar architecture with different learning algorithms to analyse the semantics of multimedia content. These two approaches offer the best performance on the TRECVID2005 conference. Both approaches combine the high-dimensional sparse text features and the low-dimensional dense features on the same feature vector. This might represent a problem for the optimization procedure because the information present on each dimension can be very different. Ideally each dimension should contain the same amount of information and the data density/sparseness should be similar across the entire feature space. The first step of our framework aims at finding this optimal trade-off point by compressing the text feature space dimension and by expanding the visual feature space dimension.

3. OPTIMAL DATA REPRESENTATION

In the problem addressed in this paper a collection of d documents is defined as the set

$$\mathcal{D} = \left\{ \left(X^{\langle 1 \rangle}, W^{\langle 1 \rangle} \right), \left(X^{\langle 2 \rangle}, W^{\langle 2 \rangle} \right), \ldots, \left(X^{\langle d \rangle}, W^{\langle d \rangle} \right) \right\}, \quad (1)$$

where each document i is identified by the pair $\left(X^{\langle i \rangle}, W^{\langle i \rangle} \right)$ corresponding to the document's features and its annotations. The feature vector $X^{\langle i \rangle}$ is decomposed into a text feature vector $T^{\langle i \rangle}$ and a visual feature vector $V^{\langle i \rangle}$, and the binary elements of the vector $W^{\langle i \rangle}$ indicate the presence of a given keyword from the vocabulary of L keywords in a document

$$X^{\langle i \rangle} = \left[T^{\langle i \rangle}, V^{\langle i \rangle} \right], \quad W^{\langle i \rangle} = \left[w_1^{\langle i \rangle}, \ldots, w_L^{\langle i \rangle} \right]. \quad (2)$$

Additionally, the elements of the n dimensional text feature vector and the m dimensional visual feature vector are real values (not discrete or only positives values)

$$T^{\langle i \rangle} = \left[t_1^{\langle i \rangle}, \ldots, t_n^{\langle i \rangle} \right], \quad V^{\langle i \rangle} = \left[v_1^{\langle i \rangle}, \ldots, v_m^{\langle i \rangle} \right]. \quad (3)$$

As discussed in the introduction, text features are very different from visual features. Processing a joint feature-space with both text and visual features would require a generic algorithm that could lead to lower indexing precisions. Moreover, because we are also targeting single-modality and multi-modality information indexing, we process each feature-space individually with algorithms adequate to the specific feature-space characteristics. To create a unique feature space where keywords are optimally represented, we transform both original feature spaces into an optimal unique feature space with a transformation

$$\mathrm{F}\left(T^{\langle i \rangle}, V^{\langle i \rangle} \right) = \left[\mathrm{F}^S\left(T^{\langle i \rangle} \right), \quad \mathrm{F}^D\left(V^{\langle i \rangle} \right) \right], \quad (4)$$

where F^S is the multivariate function that transforms sparse feature spaces (text) and F^D is the multivariate function that transforms dense feature spaces (images). The resulting function F is the simple concatenation of the other two transformations.

3.1 The MDL Principle

The transformations F^S and F^D change the representation of the original data into a different representation of the data. As we will see transformations F^S and F^D have different strategies to handle text features and visual features. However, in both cases there is the problem of selecting the optimal transformation from the large number of possible transformations and their varying complexities. In this section we answer questions like "*how many text features?*" and "*how many visual clusters?*" that are usually addressed by some heuristic method. We employ a minimum description length criterion [32], to infer the optimal representation of each feature space as follows.

When changing the representation of the data we compute a candidate transformation F^* that carries an expected error of the data on the new representation expressed with the squared-error loss, see [14]:

$$\begin{aligned} \mathrm{Err}(x) &= E\left[\left(1 - F^*(x) \right)^2 \middle| X = x \right] \\ &= \text{Irreducible Error} + \text{Bias}^2 + \text{Variance}. \end{aligned} \quad (5)$$

The first term is the variance of the modelled process and cannot be avoided. The second term measures the difference between the true mean of the process and the estimated mean. The third term is the variance of the estimated model around its mean. The more complex we make the candidate transformation F^* the lower the bias but the higher the variance. Equation (5) expresses the transformation bias-variance trade-off: simple transformations can only represent the training data's coarse details (high bias) causing a high prediction error (low variance) because the transformation ignores important aspects of the data structure; complex transformations can represent training data structures in great detail (lower bias) but the prediction error increases (in variance) because the transformation do not generalise to other data.

The MDL principle finds the transformation F^* that achieves the best trade-off between the feature space dimensionality and data representation. The solution is the transformation that minimises the description length,

$$\mathrm{DL}\left(\mathcal{D}, \mathrm{F}^*\right) = -\mathrm{DL}\left(\mathcal{D} \mid \mathrm{F}^*\right) - \mathrm{DL}\left(\mathrm{F}^*\right), \tag{6}$$

which is the description length needed to represent the data $\mathcal{D}$ in terms of a candidate transformation F^* : the description length of the transformation F^* itself, plus the description length needed to represent the data $\mathcal{D}$ on the new feature space.

3.2 Sparse Feature Spaces: Text Data

This section starts by describing the text features that were extracted from the collection's documents and then we present how to transform them to obtain an optimal feature space. Following traditional information retrieval text processing techniques [39] we remove stop words and, following Joachims [19], remove rare words from the text corpus (to avoid over-fitting). After this, the Porter stemmer [31] reduces words to their morphological root. The terms obtained by this process are weighted by their inverse document frequency [33],

$$\mathrm{IDF}\left(t_i\right) = \log\left({}^{d}\!/_{\mathrm{DF}\left(t_i\right)}\right), \tag{7}$$

where d is the number of documents in the collection and $\mathrm{DF}\left(t_i\right)$ is the number of documents containing the term t_i .

Text features are high-dimensional sparse data, which pose some difficulties to parametric generative models because each parameter receives little support from the data. In discriminative models one observes over-fitting effects because the data representation might be too optimistic by leaving out a lot of the underlying data structure information.

To find an optimal representation we define the F^{S} transformation, which reduces the number of dimensions of a sparse space with n dimensions into an optimal space with k_s dimensions, as

$$\mathrm{F}^s\left(t_1^{\langle i\rangle}, \ldots, t_n^{\langle i\rangle}\right) = \begin{bmatrix} \mathrm{f}_1^s\left(t_1^{\langle i\rangle}, \ldots, t_n^{\langle i\rangle}\right) \\ \vdots \\ \mathrm{f}_{k_s}^s\left(t_1^{\langle i\rangle}, \ldots, t_n^{\langle i\rangle}\right) \end{bmatrix}, \quad k_s \ll n\,. \tag{8}$$

3.2.1 *Term Selection*

To reduce the dimensions we rank the terms $t_0, \ \ldots \ , t_n$ by their importance for the problem classes and select the most important ones. The criterion to rank the terms (or dimensions) is the mutual information (referred to as information gain) [32], expressed by

$$\mathrm{MU}\left(w_i, t_j\right) = \sum_{w_i}\sum_{t_j} P\left(w_i, t_j\right) \log \frac{P\left(w_i, t_j\right)}{P\left(w_i\right) P\left(t_j\right)} \tag{9}$$

For each term t_j, the criterion measures the common entropy between a given query keyword entropy $H\left(w_i\right)$ and the query keyword entropy given a term t_j, $H\left(w_i \mid t_j\right)$. Yang and Pedersen [42] and Gorman [13] have shown experimentally that this is one of the best criteria for feature selection.

3.2.2 *Feature Space Selection*

With the terms ranked by their amount of entropy shared with the query keywords, we can select the most relevant terms by using the minimum description length criterion:

$$\mathrm{DL}\left(\mathcal{D}, M_\theta\right) = -\log p\left(T \mid F^S\right) + \frac{k_s}{2}\log d. \tag{10}$$

The criterion measures the trade-off between the likelihood of the data $\mathcal{D}$ for the model M_θ and the model complexity. The MDL criterion is designed "*to achieve the best compromise between likelihood and … complexity relative to the sample size*", [4]: it selects automatically the optimal feature space representation that can be obtained with an average mutual information measure.

3.3 Dense Feature Spaces: Visual Data

We now describe the visual features that were extracted from the collection's documents and then present the transformation to obtain the optimal feature space. The low-level features that we use in our implementation are a Marginal HSV colour feature [29] with 12 dimensions, a Gabor texture feature [16] with 16 dimensions, and a Tamura texture feature [16] with 3 dimensions. Images are segmented into 3 by 3 parts (9 tiles) before extracting the low-level features.

Our visual feature spaces are dense and low-dimensional spaces: hence, keyword data may overlap increasing class cross-interference. This means that the discrimination between keywords is difficult and the estimation of a density model is also less effective due to keyword data overlapping. One solution is to expand the original feature space into a higher-dimensional feature space where keywords data overlap is minimal. Thus, we define F^D as the transformation that increases the number of dimensions of a dense space with m dimensions into an optimal space with k_d dimensions:

$$\mathrm{F}^D\left(v_1^{\langle i\rangle}, \ldots, v_m^{\langle i\rangle}\right) = \begin{bmatrix} \mathrm{f}_1^D\left(v_1^{\langle i\rangle}, \ldots, v_m^{\langle i\rangle}\right) \\ \vdots \\ \mathrm{f}_{k_d}^D\left(v_1^{\langle i\rangle}, \ldots, v_m^{\langle i\rangle}\right) \end{bmatrix}, \quad k_d \gg m \tag{11}$$

3.3.1 *Visual Codebook*

Unlike most expansion techniques that use a predefined expansion we learn the expansion function F^D by exploring the natural structure of the data. The expansion function F^D is determined by estimating a density model of the entire dataset to capture its structure in the form of clusters and use each cluster as a new dimension. This strategy is similar to probabilistic latent semantic indexing [15] in the sense that we are estimating a canonical representation of the feature space.

The cluster density models of each visual feature space are computed with an expectation-maximization (EM) algorithm which fits a GMM to the data. The expression of a GMM is

$$p(x) = p(x \mid \theta_n) = \sum_{m=1}^{k_d} \alpha_m p\left(x \mid \mu_m, \sigma_m^2\right), \tag{12}$$

where k_d is the number of Gaussians (clusters), x is the low-level visual features, and θ_n represents the complete set of model parameters with component means μ_m, covariances σ_m^2, and priors α_m. The priors have the convexity constraint $\alpha_1, \ldots, \alpha_m \geq 0$ and $\sum \alpha_m = 1$. The cluster density model is estimated with an EM algorithm, which forces each cluster to model a particular and different structural aspect of the data. Since the algorithmic nature of EM reduces the cross-interference between clusters each cluster will be a new dimension of the resulting feature space. To obtain several models with different numbers of clusters (different model complexities) we estimate a hierarchal set of density models (GMMs). We developed a C++ implementation of the modified expectation-maximization algorithm proposed by Figueiredo and Jain in [12]. With minor modifications this algorithm responds to our needs, see [23]. It starts with a number of clusters much larger than the true number of clusters and deletes clusters as they get little support data or when they become singularities. Once a model is fitted, the smallest cluster is deleted and the modified EM algorithm continues with that model as a seed for estimating the next hierarchal level. The result is a hierarchy of GMMs (and equivalently a set of F^D candidate transformations) with different number of clusters (resulting dimensions).

3.3.2 *Feature Space Selection*

Once we have learned a hierarchal set of density models, we let the minimum description length criterion select automatically the density model (transformation F^D) that has the information-theoretic optimal number of clusters (dimensions):

$$\mathrm{DL}\left(\mathcal{D}, \mathrm{F}^D(\cdot)\right) = -\log p\left(V \mid \mathrm{F}^D\right) + \frac{k_D}{2}\log d \tag{13}$$

Note that this process differs from probabilistic latent semantic indexing [15] in the application of MDL criterion to select the optimal number of clusters and the creation of hierarchical models.

4. MAXIMUM ENTROPY MODEL

Maximum entropy modelling is a statistical learning technique that has been applied to a great variety of fields, e.g. natural language processing [5], text classification [28], image annotation [18]. Maximum entropy is used in this paper to model query keywords in the optimal feature space that was discussed in the previous section. As is shown in [30] maximum entropy models have an exponential (or log-linear) form

$$P(w_t \mid T, V) = \frac{1}{\mathrm{Z}(T,V)} e^{\beta_{w_t} \cdot \mathrm{F}(T,V)}, \tag{14}$$

where $\mathrm{F}(T,V)$ is the feature vector, β_{w_t} is the weight vector for keyword w_t, and $\mathrm{Z}(T,V)$ is a normalising factor to ensure a proper probability.

We implemented the binomial model, where one class is always modelled relatively to all other classes, and not a multinomial which would impose a model that does not reflect the reality of the problem. The multinomial model implies that events are exclusive and in our problem keywords are not always exclusive. For this reason, the binomial model is the correct choice for the problem at hand because documents can have more than one keyword assigned.

4.1 Over-fitting control: Gaussian Prior

As discussed by Nigan et al [28] and Chen and Rosenfeld [8], maximum entropy models may suffer from over-fitting. This is usually because features are high-dimensional and sparse meaning that the weights can easily push the model density towards some particular training data points. Zhang and Oles [44] have also presented a study on the effect of different types of regularization on logistic regression. Their results indicate that with the adequate cost function (regularization), precision results are comparative to SVMs with the advantage of rendering a probabilistic density model.

The MDL criterion already addresses this problem by selecting the optimal space complexity. Another more efficient way of tackling maximum entropy over-fitting is to set a prior on the weights. As suggested in [28] and [8] we use a Gaussian prior with mean zero and σ^2 variance to prevent the optimization procedure from over-fitting.

4.2 Large-Scale ML Estimation

To estimate the maximum entropy model the weights β_{w_t} are the only variables that need to be computed by minimizing the log-likelihood of the above model over the entire dataset

$$\beta_{w_t} = \arg\min_{\beta_{w_t}} \sum_{i \in \mathcal{D}} l\left(\beta_{w_t}\right), \tag{15}$$

where $l\left(\beta_{w_t}\right)$ is the log-likelihood function, and $\mathcal{D}$ is the entire training set. As discussed previously we use a Gaussian prior to reduce the over-fitting effect. Thus the log-likelihood function for a binomial logistic model becomes

$$l(\beta) = \sum_{i \in \mathcal{D}} \log\left(\frac{e^{\beta_{w_t} \cdot \mathrm{F}\left(T^{\langle i \rangle}, V^{\langle i \rangle}\right)}}{\mathrm{Z}\left(T^{\langle i \rangle}, V^{\langle i \rangle}\right)}\right) - \frac{\beta_{w_t}{}^{T} \beta_{w_t}}{2\sigma^2}, \tag{16}$$

where $w_t^{(i)}$ is 1 if the image i has the keyword w_t and 0 otherwise, $x^{(i)}$ is the low-level visual features of the image i, and σ^2 is the Gaussian prior variance. Thus, maximum log-likelihood model estimation is computed with a quasi-Newton algorithm that finds the solution to Equation (15) by finding the root of the first derivative of Equation (16):

$$\frac{\partial l(\beta)}{\partial \beta} = \sum_{i \in \mathcal{D}} \mathrm{F}\left(x^{(i)}\right)\left(w_t{}^{(i)} - p\left(w_t \mid x^{(i)}, \beta\right)\right) - \frac{\beta}{\sigma^2}. \quad (17)$$

Newton algorithms need the Hessian matrix to drive the algorithm into a local maximum solution. The computation of the Hessian matrix is very complex because the feature space might have up to around 10,000 dimensions producing the computation of a 10,000×10,000 on each iteration. Thus, algorithms that compute approximations to the Hessian matrix are ideal for the problem at hand. The limited-memory BFGS algorithm proposed by Liu and Nocedal [22] is one of such algorithms that "*use curvature information from only the most recent iterations to construct the Hessian approximation. Curvature information from earlier iterations, which is less likely to be relevant to the actual behaviour of the Hessian at the current iteration, is discarded in the interest of saving storage*". Malouf [25] has compared several optimisation algorithms for maximum entropy and found the limited-memory BFGS algorithm to be the best one. We use the implementation provided by Liu and Nocedal [22].

5. EVALUATION

To evaluate our information-theoretic framework we tested it on a text dataset, an image dataset, and a text and image dataset. The following sections will present these datasets, a baseline classifier, the experiments, and the results of the evaluation.

5.1 Datasets

Reuters-21578: This is a widely used text dataset which allows comparing our results with others in the literature. Each document is composed by a text corpus, a title (which we ignore), and labelled categories. This dataset has several possible splits and we have used the ModApte split which contains 9,603 training documents and 3,299 test documents (the same evaluation setup used in [19, 26, 28, 44]). Terms appearing less than 3 times were removed. Only labels with at least 1 document on the training set and the test set were considered leaving us with 90 labels. After these steps we ended with 7,770 labelled documents for training.

Corel Images: This dataset was compiled by Duygulu et al. [10] from a set of COREL Stock Photo CDs. The dataset has some visually similar concepts (jet, plane, Boeing), and some concepts have a limited number of examples (10 or less). In their seminal paper, the authors acknowledge that fact and ignored the classes with these problems. In this paper we use the same setup as in [7, 11, 17, 20, 43], which differ slightly from the one used in the dataset original paper, [10]. The retrieval evaluation scenario consists of a training set of 4,500 images and a test set of 500 images. Each image is annotated with 1-5 keywords from a vocabulary of 371 keywords. Only keywords with at least 2 images in the test set were evaluated which reduced the number of vocabulary to 179 keywords.

TRECVID2005: To test the algorithm on a multi-modal collection of documents we used TRECVID2005: each document (a video shot) has text (from speech), images (keyframes) and the labels from vocabulary of 39 standard keywords rather than the full 400 LSCOM concepts. Since only the training set is completely labelled, we randomly split the training English videos to use as train and test. We considered each document to be a keyframe plus the text within a window of 6 seconds, and the retrieval evaluation was done at the document (shot) level. We did not consider the non-English data because that would require more time/processing power and the English ASR in these cases are too noisy (the ASR obtained from speech recognition followed by machine translation).

5.2 Baseline Naïve Bayes Model

The naïve Bayes text classifier results from the direct application of Bayes law and from the use of strong independence assumptions between terms in a document. As discussed by McCallum and Nigam [26], a document can be represented as an event model of term presence or term count, leading to the choice of a binomial or multinomial model respectively. We choose the multinomial distribution, as the binomial distribution is too limiting given the probabilistic nature of our problem. The description of the naïve Bayes implementation used in our experiments is in [24].

5.3 Experiments and Results

We run retrieval experiments by ranking documents for each keyword and computing the corresponding average precision. The mean of the results for all keywords, the mean average precision, is plotted on figures against the dimension of the feature spaces. The mark indicates the results with the feature space chosen with the minimum description length. The regularization parameter was chosen by cross-validation. The graphs also compare the maximum entropy framework to a baseline naïve Bayes model. The low-level visual features are: Marginal HSV colour feature [29] with 12 dimensions; Gabor texture feature [16] with 16 dimensions; Tamura texture feature [16] with 3 dimensions. Images are segmented into 3 by 3 parts (9 tiles) before extracting the low-level features. Text features are processed as described previously.

Results in the Reuters dataset in Figure 1 show that after some number of terms (space dimension) precision does not increase because the information carried by these terms are already present on the previous ones. It is interesting to note that the MDL point is slightly below the best value because the number of samples is too small and does not favour more complex models (see the MDL expression).

Figure 2 shows the retrieval results versus the data representation complexity for the Corel images dataset. Each point in the curve is obtained by concatenating different colour and texture representations with the same number of dimensions (e.g. 100 dimensions for colour plus 100 dimensions for texture).

The precision stabilises after a certain space dimension because the new dimensions being added to the feature space do not bring any original information (the same phenomenon shows on the Reuters dataset). The MDL point is not on top of the curve because it corresponds to the concatenation of the best representation of the colour features plus the best representation of the texture features. Note that in the Reuters case the MDL point is on top of the curve because there is only a single feature space.

With the TRECVID dataset we tested our statistical modelling framework on data with both dense and sparse data. Figure 3 shows the results and it is possible to observe the same phenomenon that we observed on the other datasets. Note also that the difference between naïve Bayes and maximum entropy is not big which we believe is due to the fact of the increasing number of parameters to estimate.

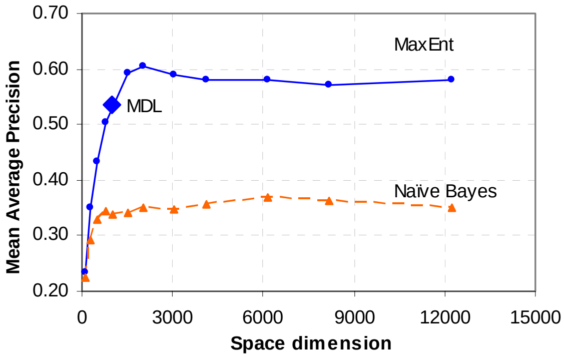

Figure 1 – Retrieval results on Reuters-21578.

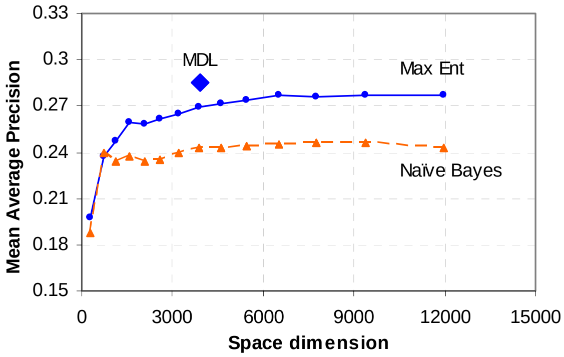

Figure 2 – Retrieval results on Corel Images.

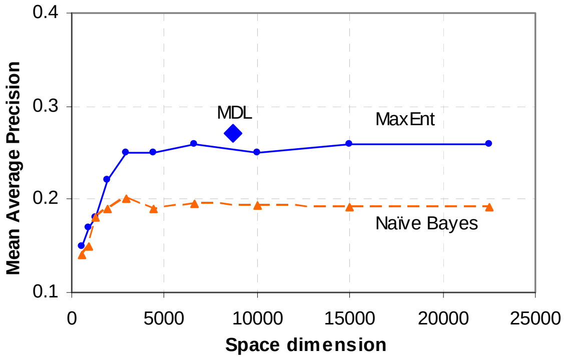

Figure 3 – Multimodal retrieval results on TRECVID2005.

6. CONCLUSIONS

We propose an information-theoretic framework for semantically indexing text, images and multimedia information. The text, image, and multimedia models are a multi-modal representation of a query keyword on an information retrieval system. Text and visual features are transformed via information theory related techniques (average mutual information and clustering) into an optimal representation of the original data with the MDL criterion. Finally, the query keywords are represented as a maximum entropy model regularised with a Gaussian prior and estimated with a quasi-Newton algorithm.

Precision. The performed experiments show that our framework offers a performance in the same range as other state-of-the-art algorithms. Text and image results are quite good while multimodal experiments were affected by the noise present on the speech text and by the higher number of parameters to estimate. It was not surprising to see that maximum entropy attains better results than naïve Bayes at the expense of a higher learning cost.

Model selection. The MDL criterion selects the optimal complexity of a model that faithfully represents data for the given number of samples. It does not necessarily select the model that achieves the best results. In some situations cross-validation might select a model which produces better results for two reasons: (1) sometimes the assumption that data was generated by a random process is too weak (there's a strong bias between the train and test set) and (2) the number of samples is too low leading the MDL criterion to select a simpler model.

Precision vs space dimensions. We use MDL to select the optimal representation of each individual feature space and not the optimal representation of all feature spaces together. Individual feature spaces contain redundant information that already exists on some other feature space. This means that after merging all new data representations, there will be dimensions containing related information. To solve this problem one would have to employ some type of greedy search algorithm that gathers a feature space with only the useful dimensions, discarding the redundant ones. Note that this is related to the way we design new representations of data and not to the way we assess them (in our case with MDL).

Learning scalability. The high computational cost of the learning process resides on the clustering of the visual feature space and on the quasi-Newton algorithm. These learning procedures are usually done offline and they aim at estimating the keyword model with the minimum complexity possible which results in a simple model with a high inference scalability.

Indexing scalability. In contrast to most maximum entropy models that have a set of feature functions specific for each keyword, we have a unique set of features to compute all keywords probabilities. Obviously, this results in a low complexity indexing algorithm which is crucial for large-scale search engines. The clustering of the visual feature space contributes to this reduction on the computational complexity: apart from its hierarchical nature, it pursues the same objective as probabilistic latent semantic indexing [15], which is to approximate the SVD canonical representation of a feature space.

7. REFERENCES

[1] A. Amir, J. O. Argillander, M. Campbell, A. Haubold, G. Iyengar, S. Ebabdollahi, F. Kang, M. Naphade, A. Natsev, J. R. Smith, J. Tesic, T. Volkmer, "IBM Research TRECVID-2005 video retrieval system," TREC Video Retrieval Evaluation Workshop, Gaithersburg, MD, USA, 2005.

[2] J. Argillander, G. Iyengar, H. Nock, "Semantic annotation of multimedia using maximum entropy models," IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing, Philadelphia, PA, 2005.

[3] K. Barnard, D. A. Forsyth, "Learning the semantics of words and pictures," Int'l Conf. on Computer Vision, Vancouver, Canada, 2001.

[4] A. Barron, T. Cover, "Minimum complexity density estimation," *IEEE Trans. on Information Theory*, vol. 37, pp. 1034-1054, 1991.

[5] A. Berger, S. Pietra, V. Pietra, "A maximum entropy approach to natural language processing," Computational Linguistics, 1996.

[6] D. Blei, M. Jordan, "Modeling annotated data," ACM SIGIR, Toronto, Canada, 2003.

[7] G. Carneiro, N. Vasconcelos, "Formulating semantic image annotation as a supervised learning problem," IEEE Conf. on Computer Vision and Pattern Recognition, San Diego, CA, USA, 2005.

[8] S. F. Chen, R. Rosenfeld, "A Gaussian prior for smoothing maximum entropy models," Carnegie Mellon University, Pittsburg, PA February 1999.

[9] T. M. Cover, J. A. Thomas, *Elements of information theory*: John Wiley & Sons, 1991.

[10] P. Duygulu, K. Barnard, N. de Freitas, D. Forsyth, "Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary," European Conf. on Computer Vision, Copenhagen, Denmark, 2002.

[11] S. L. Feng, V. Lavrenko, R. Manmatha, "Multiple Bernoulli relevance models for image and video annotation," IEEE Conf. on Computer Vision and Pattern Recognition, Cambridge, UK, 2004.

[12] M. Figueiredo, A. K. Jain, "Unsupervised learning of finite mixture models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 381-396, 2002.

[13] G. Forman, "An Extensive Empirical Study of Feature Selection Metrics for Text Classification," *Machine Learning Research*, pp. 1289-1305, 2003.

[14] T. Hastie, R. Tibshirani, J. Friedman, *The elements of statistical learning: Data mining, inference and prediction*: Springer, 2001.

[15] T. Hofmann, "Probabilistic latent semantic indexing," ACM SIGIR, Berkeley, CA, USA, 1999.

[16] P. Howarth, S. Rüger, "Evaluation of texture features for content-based image retrieval," Int'l Conf. on Image and Video Retrieval, Dublin, Ireland, 2004.

[17] J. Jeon, V. Lavrenko, R. Manmatha, "Automatic image annotation and retrieval using cross-media relevance models," ACM SIGIR, Toronto, Canada, 2003.

[18] J. Jeon, R. Manmatha, "Using maximum entropy for automatic image annotation," Int'l Conf on Image and Video Retrieval, Dublin, Ireland, 2004.

[19] T. Joachims, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features," European Conf. on Machine Learning, 1998.

[20] V. Lavrenko, R. Manmatha, J. Jeon, "A model for learning the semantics of pictures," Neural Information Processing System Conf., Vancouver, Canada, 2003.

[21] S. Lazebnik, C. Schmid, J. Ponce, "A maximum entropy framework for part-based texture and object recognition," Int'l Conf. on Computer Vision, Beijing, China, 2005.

[22] D. C. Liu, J. Nocedal, "On the limited memory method for large scale optimization," *Mathematical Programming B*, vol. 45, pp. 503-528, 1989.

[23] J. Magalhães, S. Rüger, "Logistic regression of generic codebooks for semantic image retrieval," Int'l Conf. on Image and Video Retrieval, Phoenix, AZ, USA, 2006.

[24] J. Magalhães, S. Rüger, "High-Dimensional Visual Vocabularies for Image Retrieval," ACM SIGIR, Amsterdam, Holland, 2007.

[25] R. Malouf, "A comparison of algorithms for maximum entropy parameter estimation," Sixth Conf. on Natural Language Learning, Taipei, Taiwan, 2002.

[26] A. McCallum, K. Nigam, "A comparison of event models for naive Bayes text classification," AAAI Workshop on Learning for Text Categorization, 1998.

[27] M. R. Naphade, T. S. Huang, "A probabilistic framework for semantic video indexing filtering and retrieval," *IEEE Trans. on Multimedia*, vol. 3, pp. 141-151, 2001.

[28] K. Nigam, J. Lafferty, A. McCallum, "Using Maximum Entropy for Text Classification," IJCAI - Workshop on Machine Learning for Information Filtering, Stockholm, Sweden, 1999.

[29] M. J. Pickering, D. Heesch, R. O'Callaghan, S. Rüger, D. Bull, "Video retrieval using global features in keyframes," TREC Text Retrieval Conf. , Gaithersburg, USA, 2002.

[30] S. D. Pietra, V. D. Pietra, "Inducing features of random fields," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 380-393, 1997.

[31] M. F. Porter, "An algorithm for suffix stripping," *Program*, vol. 14, pp. 130-137, 1980.

[32] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, pp. 465-471, 1978.

[33] G. Salton, C. Buckley, "Term weighting approaches in automatic text retrieval," *Information Processing and Management*, vol. 24, pp. 513-523, 1988.

[34] C. G. M. Snoek, J. C. v. Gemert, T. Gevers, B. Huurnink, D. C. Koelma, M. v. Liempt, O. d. Rooij, K. E. A. v. d. Sande, F. J. Seinstra, A. W. M. Smeulders, A. H. C. Thean, C. J. Veenman, M. Worring, "The MediaMill TRECVID 2006 Semantic Video Search Engine," TREC Video Retrieval Evaluation Workshop, Gaithersburg, MD, USA, 2006.

[35] C. G. M. Snoek, M. Worring, J.-M. Geusebroek, D. C. Koelma, F. J. Seinstra, A. W. M. Smeulders, "The semantic pathfinder: using an authoring metaphor for generic multimedia indexing," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 28, pp. 1678-1689, 2006.

[36] A. Vailaya, M. Figueiredo, A. K. Jain, H. J. Zhang, "Image classification for content-based indexing," *IEEE Trans. on Image Processing*, vol. 10, pp. 117-130, 2001.

[37] T. Westerveld, A. P. de Vries, "Experimental result analysis for a generative probabilistic image retrieval model," ACM SIGIR, Toronto, Canada, 2003.

[38] T. Westerveld, A. P. de Vries, T. Ianeva, L. Boldareva, D. Hiemstra, "Combining information sources for video retrieval," TREC Video Retrieval Evaluation Workshop, Gaithersburg, MD, USA, 2003.

[39] Y. Yang, "An Evaluation of Statistical Approaches to Text Categorization," *Information Retrieval*, pp. 69-90, 1999.

[40] Y. Yang, C. G. Chute, "An example-based mapping method for text categorization and retrieval," *ACM Trans. on Information Systems*, vol. 13, pp. 252-277, 1994.

[41] Y. Yang, X. Liu, "A re-examination of text categorization methods," SIGIR, 1999.

[42] Y. Yang, J. O. Pedersen, "A Comparative Study on Feature Selection in Text Categorization," Int'l Conf. on Machine Learning, Nashville, Tennessee, USA, 1997.

[43] A. Yavlinsky, E. Schofield, S. Rüger, "Automated image annotation using global features and robust nonparametric density estimation," Int'l Conf. on Image and Video Retrieval, Singapore, 2005.

[44] T. Zhang, F. J. Oles, "Text Categorization Based on Regularized Linear Classification Methods," *Information Retrieval*, pp. 5-31, 2001.