Efficient Maximum Entropy Training for Statistical Object Recognition

Daniel Keysers, Franz Josef Och, and Hermann Ney
keysers@cs.rwth-aachen.de

Lehrstuhl für Informatik VI, Computer Science Department
RWTH Aachen – University of Technology, D-52056 Aachen, Germany

Supervisor:	Prof. Dr.-Ing H. Ney
Type:	PhD thesis
GI subjects:	image understanding (1.0.4), machine learning (1.1.3)

Abstract
In statistical pattern recognition, we use probabilistic models within the task of assigning observations to one of a set of predefined classes, like e.g. images of handwritten digits to one of the classes '0' to '9'. The principle of maximum entropy is a powerful framework that can be used to estimate class posterior probabilities for pattern recognition tasks. It is a conceptually simple model that allows to estimate a large number of free parameters reliably. We show how to apply this framework to object recognition and compare the results to other state-of-the-art approaches in experiments with the well known US Postal Service handwritten digits recognition task. We also introduce a simple but effective heuristic method for speeding up the algorithms used to determine the model parameters.

1 Introduction

In pattern recognition, our goal is to assign an observation represented as a feature vector $x \in \mathbb{R}^D$ to one of a set of predefined classes $\{1, \ldots, K\}$. For example, in handwritten digit classification, the feature vectors may represent the greyvalues within a digitized image scanned from a postal envelope. In this case, we want to determine which of the $K = 10$ classes labeled with '0' to '9' the image belongs to. To classify an observation x, we use Bayes' decision rule [1]:

$$x \quad \longmapsto \quad r(x) = \operatorname*{argmax}_k \{p(k|x)\} = \operatorname*{argmax}_k \{p(k)\ p(x|k)\}$$

Here, $p(k|x)$ is the class posterior probability of class $k \in \{1, \ldots, K\}$ given the observation x, $p(k)$ is the a priori probability, $p(x|k)$ is the class conditional probability for the observation x given class k, and $r(x)$ is the decision of the classifier. Hence, Bayes' decision rule tells us to choose the class that has the highest probability given the observed information. This decision rule is known to be optimal with respect to the number of decision errors, if the correct distributions are known. This is generally not the case in practical situations, which means that we need to choose appropriate models for the distributions. In the training phase, the parameters of the distribution are estimated from a set of training data $\{(x_n, k_n)\}$, $n = 1, \ldots, N$, $k_n \in 1, \ldots, K$. If we denote by Λ the set of free parameters of the distribution, the maximum likelihood approach consists in choosing the parameters $\hat{\Lambda}$ that maximize the log-likelihood on the training data:

$$\hat{\Lambda} = \operatorname*{argmax}_\Lambda \sum_n \log p_\Lambda(x_n|k_n) \tag{1}$$

Alternatively, we can maximize the log-probability of the class posteriors,

$$\hat{\Lambda} = \operatorname*{argmax}_\Lambda \sum_n \log p_\Lambda(k_n|x_n)\ , \tag{2}$$

which is also called discriminative training, since the information of out-of-class data is used. This criterion is often referred to as mutual information criterion.

2 Maximum Entropy Modeling for Pattern Recognition

The principle of maximum entropy has origins in statistical thermodynamics, is related to information theory and has been applied to pattern recognition tasks such as language modeling and text classification. Applied to classification, the basic idea is the following: We are given information about a probability distribution by samples from that distribution (training data). Now, we choose the distribution such that it fulfills all the constraints given by that information, but otherwise has the highest possible entropy. (This inherently serves as regularization to avoid overfitting.) It can be shown that this approach leads to so-called log-linear models for the distribution to be estimated.

Consider a set of so-called feature functions $\{f_i\}, i = 1, \ldots, I$ that are supposed to compute 'useful' information for classification:

$$f_i \quad : \quad \mathbb{R}^D \times \{1, \ldots, K\} \longrightarrow \mathbb{R} \quad : \quad (x, k) \longmapsto f_i(x, k)$$

Now, the maximum entropy principle consists in maximizing the entropy

$$\max_{p(k|x)} \Big\{ -\sum_n \sum_k p(k|x_n) \log p(k|x_n) \Big\}$$

over all possible distributions with the requirements:

- normalization constraint for each observation x: $\sum_k p(k|x) = 1$
- feature constraint for each feature i: $\sum_n \sum_k p(k|x_n) f_i(x_n, k) = \sum_n f_i(x_n, k_n) =: N_i$

It can be shown that the resulting distribution has the following log-linear or exponential functional form:

$$p_\Lambda(k|x) \quad = \quad \frac{\exp\left[\sum_i \lambda_i f_i(x, k)\right]}{\sum_{k'} \exp\left[\sum_i \lambda_i f_i(x, k')\right]}, \quad \Lambda = \{\lambda_i\}. \tag{3}$$

Interestingly, it can also be shown that the stated optimization problem is convex and has a unique global maximum. Furthermore, this unique solution is also the solution to the following dual problem: Maximize the log-probability (2) on the training data using the model (3). A second desirable property of the discussed model is that effective algorithms are known that compute the global maximum of the log-probability (2) given a set of training data. These algorithms fall into two categories: On the one hand, we have an algorithm known as generalized iterative scaling (GIS, [2]) and related algorithms that can be proven to converge to the global maximum. On the other hand, due to the convex nature of the criterion (2), we can also use general optimization strategies as e.g. conjugate gradient methods. The crucial problem in maximum entropy modeling is the choice of the appropriate feature functions $\{f_i\}$.

3 Parameter Estimation and Heuristic Speed-up

The GIS algorithm [2] proceeds as follows to determine the free parameters of the model (3). First, we choose an initial parameter set $\Lambda^{(0)} = \{\lambda_i^{(0)}\}$. Then, for each iteration $m = 1, \ldots, M$ the parameters are updated according to

$$\lambda_i^{(m)} = \lambda_i^{(m-1)} + \Delta\lambda_i^{(m)} = \lambda_i^{(m-1)} + \frac{1}{F} \log \frac{N_i}{Q_i^{(m)}}, \quad \text{where } Q_i^{(m)} := \sum_n \sum_k p_{\Lambda^{(m)}}(k|x_n) f_i(x_n, k)$$

and F is a constant depending on the training data. (Because of space limitations we refer to the references for details on the computation of the updates.) This computation is expensive as it requires one pass over the training data to determine the probabilities $p(k|x_n)$ for each class k, while summing values for each feature f_i. Furthermore, the convergence of the algorithm may take many iterations for complex distributions, resulting in a high computational cost.

Interestingly, it can be observed for different tasks that consecutive update vectors tend to be similar to each other especially for increasing numbers of iterations. This similarity can be measured by the cosine of the angle between two consecutive update vectors. Now, we can assume that in regions where the cosine is close to one (i.e. the vectors point into very similar directions in the vector space of possible parameter sets), the update vector can be multiplied by a factor greater than one. This yields a faster convergence of the algorithm (i.e. convergence within a smaller number of iterations). This procedure implies that we cannot theoretically guarantee convergence of the algorithm any more, but experiments show possible speed-ups up to 20 times faster convergence. Furthermore, we can ensure convergence of the algorithm by observing the log-probability (2) on the training data in each iteration and falling back to the conventional update strategy if it decreases.

Note that there exists an enhanced version of the GIS algorithm known as improved iterative scaling [3] which in most cases converges faster. The speed-up method presented here may also be applied effectively to this improved version. This is especially true in the case where feature normalization (see below) is applied, as in that case both algorithms are identical.

Table 1: Summary of results for the USPS corpus (error rates, [%]).
*: training set extended with 2,400 machine-printed digits

method	reference	ER[%]
human performance	[SIMARD et al. 1993] [4]	2.5
relevance vector machine	[TIPPING et al. 2000] [5]	5.1
neural net (LeNet1)	[LECUN et al. 1990] [4]	4.2
invariant support vectors	[SCHÖLKOPF et al. 1998] [6]	3.0
neural net + boosting	[DRUCKER et al. 1993] [4]	*2.6
tangent distance	[SIMARD et al. 1993] [4]	*2.5
extended tangent distance	[KEYSERS et al. 2000] [7]	2.4

4 Experiments and Results

We performed experiments on the well known US Postal Service handwritten digit recognition task (USPS). It contains normalized greyscale images of handwritten digits taken from US zip codes of size 16×16 pixels. The corpus is divided into a training set of 7,291 images and a test set of 2,007 images. Reported recognition error rates for this database are summarized in Table 1.

We used the most direct features possible in the experiments, which also have an interesting relation to Gaussian Models [8]. Features of order 0, 1 and 2 are given by

$$\begin{aligned} f_k(x,k') &= \delta(k,k') \ , \\ f_{k,i}(x,k') &= \delta(k,k')\, x_i \ , \text{ and} \\ f_{k,i,j}(x,k') &= \delta(k,k')\, x_i x_j \ , \quad i \geq j \ , \end{aligned}$$

respectively, where $\delta(k,k') := 1$ if $k = k'$, and 0 otherwise denotes the Kronecker delta function. In the context of image recognition, we may call these functions appearance based image features, as they represent the image pixel values. The duplication of the features for each class is necessary to distinguish the hypothesized classes.

In most of the experiments performed we obtained better results using 'feature normalization'. This means that we enforced for each observation during training and testing that the sum of all feature values is equal to one by scaling the feature values appropriately. Thus, we obtain new feature functions $\{\tilde{f}_i\}$:

$$\forall x,k,i: \ \tilde{f}_i(x,k) = \Big(\sum_{i'} f_{i'}(x,k)\Big)^{-1} \cdot f_i(x,k)$$

In the following, we only report result obtained using feature normalization.

Table 2 shows the main results obtained in comparison to other approaches along with the number of free parameters of the respective models [8]. Taking into account the class information in training using the maximum entropy framework (i.e. switching from maximum likelihood to maximum mutual information criterion) improves the recognition accuracy for first-order features from 18.6% to 8.2% error rate. Furthermore, it can be observed that the maximum entropy models perform better for second-order features than for first-order features, which stands in contrast to the experience gained with maximum likelihood estimation of Gaussian densities[9]. Note e.g. that the maximum likelihood estimation of class specific diagonal covariance matrices already imposes problems for the USPS data as in some of the classes some of the dimensions have zero variance in the training data. Here, the maximum entropy framework offers an effective way to overcome these problems. Using the equivalent of a full class specific covariance matrix, i.e. second-order features, the

Table 2: Overview of the results obtained on the USPS corpus using maximum entropy modeling in comparison to other models (error rates, [%]). ML: maximum likelihood, MMI: maximum mutual information, *: with pooled diagonal covariance matrix.

model	training criterion	# parameters	ER[%]
Gaussian model*	ML	2 816	18.6
maximum entropy, first-order features	MMI	2 570	8.2
second-order features	MMI	331 530	5.7
nearest neighbor classifier		1 866 496	5.6

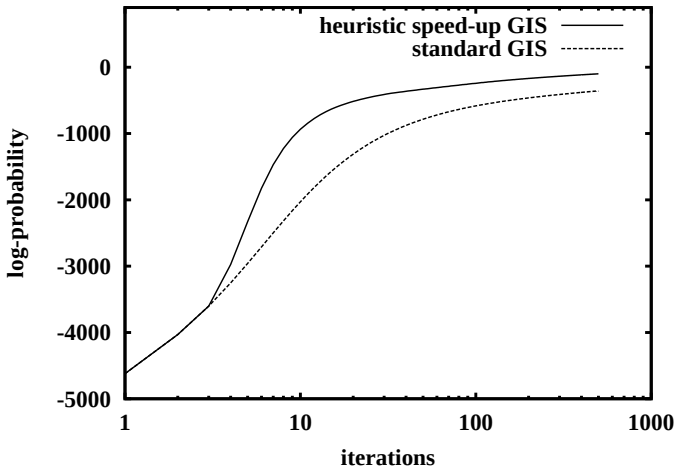

Figure 1: Log-probability (2) on the training data (USPS, first-order features) as a function of the number of iterations for standard GIS and heuristic speed-up GIS.

error rate of 5.7% approaches that of a nearest neighbor classifier, which has more than five times as many parameters.
Figure 1 shows the log-probability (2) on the training data for first-order features as a function of the number of iterations for standard GIS and heuristic speed-up GIS. It can be observed that on this data the proposed speed-up can save around 90% of training time.

5 Conclusion

We showed the use of the maximum entropy framework for object recognition and introduced a new heuristic speed-up technique for the training of maximum entropy models. The framework allows the estimation of a large number of parameters reliably and the corresponding optimization problem has a number of desirable properties. A further advantage of the maximum entropy approach is that it is easily possible to include new feature functions into the classifier.
We evaluated the approach for image object recognition using the US Postal Service handwritten digits recognition task. The best result of 5.7% error rate using second-order features is competitive with other results reported on this dataset, although approaches with significantly better performance exist. (Note that the latter are highly tuned to the specific task at hand while the maximum entropy approach is of very general nature.) The accuracy of the resulting model shows that the maximum entropy approach allows robust estimation of a large number of parameters even on this small training set, which may be a problem for approaches based on maximum likelihood.

References

[1] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley & Sons, Inc., New York, 2nd edition, 2001.
[2] J. N. Darroch and D. Ratcliff. Generalized Iterative Scaling for Log-Linear Models. *Annals of Mathematical Statistics*, 43(5):1470–1480, 1972.
[3] S. Della Pietra, V. Della Pietra, and J. Lafferty. Inducing Features of Random Fields. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19(4):380–393, April 1997.
[4] P. Simard, Y. Le Cun, J. Denker, and B. Victorri. Transformation Invariance in Pattern Recognition — Tangent Distance and Tangent Propagation. volume 1524, Springer, Heidelberg, pages 239–274, 1998.
[5] M. E. Tipping. The Relevance Vector Machine. In *Advances in Neural Information Processing Systems 12*. MIT Press, pages 332–388, 2000.
[6] B. Schölkopf, P. Simard, A. Smola, and V. Vapnik. Prior Knowledge in Support Vector Kernels. In *Advances in Neural Information Processing Systems 10*. MIT Press, pages 640–646, 1998.
[7] D. Keysers, J. Dahmen, T. Theiner, and H. Ney. Experiments with an Extended Tangent Distance. In *Proc. 15th Int. Conf. on Pattern Recognition*, volume 2, Barcelona, Spain, pages 38–42, September 2000.
[8] D. Keysers, F. J. Och, and H. Ney. Maximum Entropy and Gaussian Models for Image Object Recognition. In *22. DAGM Symposium for Pattern Recognition*, Zürich, Switzerland, September 2002. In press.
[9] J. Dahmen, D. Keysers, H. Ney, and M. O. Güld. Statistical Image Object Recognition using Mixture Densities. *J. Mathematical Imaging and Vision*, 14(3):285–296, May 2001.