Learning Preference Relations for Information Retrieval

Ralf Herbrich*, Thore Graepel†, Peter Bollmann-Sdorra*, Klaus Obermayer†
Department of Computer Science, Technical University of Berlin,
* Statistic Research Group, Sekr. FR 6-9,
† Neural Information Processing Group, Sekr. FR 2-1,
Franklinstr. 28/29, 10587 Berlin, Germany
ralfh|graepel2|bollmann|oby@cs.tu-berlin.de

Abstract

In this paper we investigate the problem of learning a preference relation from a given set of ranked documents. We show that the Bayes's optimal decision function, when applied to learning a preference relation, may violate transitivity. This is undesirable for information retrieval, because it is in conflict with a document ranking based on the user's preferences. To overcome this problem we present a vector space based method that performs a linear mapping from documents to scalar utility values and thus guarantees transitivity. The learning of the relation between documents is formulated as a classification problem on pairs of documents and is solved using the principle of structural risk minimization for good generalization. The approach is extended to polynomial utility functions by using the potential function method (the so called "kernel trick"), which allows to incorporate higher order correlations of features into the utility function at minimal computational costs. The resulting algorithm is tested on an example with artificial data. The algorithm successfully learns the utility function underlying the training examples and shows good classification performance.

Introduction

The task of supervised learning in information retrieval (IR) is mostly based on the assumption that a given document is either relevant or non-relevant. This holds for example for Rocchio's feedback algorithm (Salton 1968) and for the binary independence model (Robertson 1977) which is based on a Bayesian approach. A classification approach was adopted and as classifications were considered to be partitions on a set of objects this reduces to learning equivalence relations from examples. But there is also the view that the similarity of the documents to the query represents the importance of the documents (Salton 1989, p. 317), which in turn means that a user need implies some preference relation on the documents. In (Bollmann & Wong 1987) and (Wong, Yao, & Bollmann 1988) the idea was developed to learn a preference relation instead of an equivalence relation. The learning of preference relations reduces to a standard classification problem if pairs of objects are considered, because a binary relation can be viewed as a subset of the Cartesian product. (Wong, Yao, & Bollmann 1988) successfully applied linear classification and perceptron learning to this problem.

In this paper we consider the situation that there are more than two relevance levels and that there exist several documents with different relevance levels which all have the same description. We find that an ideal Bayesian approach leads to inconsistencies, namely to the violation of transitivity. To overcome this problem, an algorithm is developed which enforces transitivity by learning a linear mapping from document descriptions to scalar utility values based on training examples that consist of pairs of document descriptions and their preference relation. The learning procedure is based on the principle of structural risk minimization (SRM) (Vapnik 1995), which is known for its good generalization properties (for an application of SRM to document classification see (Joachims 1997)). The linear approach is generalized to include nonlinear utility functions, which are able to capture correlations between the features, by applying the so-called "kernel-trick". The paper is structured as follows: First, the learning of preference relations is formulated as a classification problem on pairs of document descriptions and the inconsistency of the Bayesian approach is demonstrated. In the following, the linear vector space model is introduced and structural risk minimization is applied for learning the weight vector. Then, this approach is generalized to include nonlinear utility functions by applying the "kernel trick". Finally, we present some numerical experiments to demonstrate the validity of the approach.

The Problem of Transitivity

Let us consider a static document space denoted by D with documents $d \in D$ being represented by feature vectors $\mathbf{d} = (d_1, d_2, \ldots, d_n)^{\cdot} \in \mathcal{D}$ where n denotes the number of the features d_k. The user determines a preference relation on the documents used for training, and generates a training set S consisting of ℓ pairs $(\mathbf{d}, \mathbf{d}')$ of document descriptions together with their relations $\mathbf{d} \bullet\!\!> \mathbf{d}'$:

$$S = \{((\mathbf{d}^{(i)}, \mathbf{d}'^{(i)}), y^{(i)})\}_{i=1}^{\ell}$$

relevance levels	no. of documents		
	$\mathbf{d}$	$\mathbf{d}'$	$\mathbf{d}''$
R_1	0	0	4
R_2	5	0	0
R_3	0	9	0
R_4	0	0	5
R_5	4	0	0

Table 1: Number of relevance assignments to 27 documents described by the feature vectors $\mathbf{d}$, $\mathbf{d}'$,and $\mathbf{d}''$

$$y^{(i)} = \begin{cases} +1 & \text{if } \mathbf{d}^{(i)} \bullet\!\!> \mathbf{d}'^{(i)} \\ -1 & \text{otherwise} \end{cases} . \quad (1)$$

Moreover, let us consider a set $\mathcal{H}$ of functions h_j, which map pairs of documents to the set $\{+1, -1\}$.

We are now in a position to formulate our problem: Given a training set S and a space of hypotheses $\mathcal{H}$, choose one function $h^* \in \mathcal{H}$ such that the risk of misclassifying further pairs (d, d') of documents is minimized. Moreover, the relation represented by h^* has to be transitive,

$$h^*(\mathbf{d}, \mathbf{d}') = h^*(\mathbf{d}', \mathbf{d}'') = +1 \Rightarrow h^*(\mathbf{d}, \mathbf{d}'') = +1 . \quad (2)$$

Our task now reduces to a classification problem. The objects the classifier has to assign to the classes $\bullet\!\!>$ and $\neg\bullet\!\!>$ are pairs $(\mathbf{d}, \mathbf{d}')$ of document descriptions. From the theory of optimal decisions in classification tasks (e.g., (Bishop 1995)) it is known, that the function h^* with minimal risk is the Bayes's optimal function:

$$h^*(\mathbf{d}, \mathbf{d}') = \begin{cases} +1 & \text{if } P(\bullet\!\!> \,|(\mathbf{d}, \mathbf{d}')) > \frac{1}{2} \\ -1 & \text{otherwise} \end{cases} . \quad (3)$$

However, the Bayesian approach for preference learning is inconsistent, because stochastic transitivity may not hold (Suppes *et al.* 1989). We will demonstrate this fact by the following example.

Let us consider a document space with 27 documents. The documents are described by three distinct feature vectors $\mathcal{D} = \{\mathbf{d}, \mathbf{d}', \mathbf{d}''\}$, which separate the document space into three sets of nine documents each, one set for each feature vector. Each document is assigned one out of five relevance levels as listed in Table 1, where $d \in R_i$ is preferred over $d' \in R_j$ iff $i < j$. Hence the choice probabilities are $P(\bullet\!\!> \,|(\mathbf{d}, \mathbf{d}')) = \frac{45}{81} = \frac{5}{9} > \frac{1}{2}$, $P(\bullet\!\!> \,|(\mathbf{d}', \mathbf{d}'')) = \frac{45}{81} = \frac{5}{9} > \frac{1}{2}$, and $P(\bullet\!\!> \,|(\mathbf{d}'', \mathbf{d})) = \frac{56}{81} > \frac{1}{2}$. This is equivalent (using (3)) to

$$h^*(\mathbf{d}, \mathbf{d}') = h^*(\mathbf{d}', \mathbf{d}'') = h^*(\mathbf{d}'', \mathbf{d}) = +1 , \quad (4)$$

which contradicts transitivity (2).

The Utility Function Approach

One way to enforce transitivity is to map each document description to a real value: $U : \mathcal{D} \mapsto \Re$. Such a value can be seen as an ordinal utility a document provides to the user (Roberts 1979). The transitivity of the relation is assured by the rule

$$\mathbf{d} \bullet\!\!> \mathbf{d}' \quad \Leftrightarrow \quad U(\mathbf{d}) > U(\mathbf{d}') , \quad (5)$$

which maps the classification problem to the problem of learning the function $U(\mathbf{d})$. Let us start by making a linear model of the function $U(\mathbf{d})$ parameterized by an n–dimensional vector $\mathbf{w} = (w_1, \ldots, w_n)$ (Wong, Yao, & Bollmann 1988):

$$U(\mathbf{d}) = \sum_{k=1}^{n} w_k d_k + b = \mathbf{w} \cdot \mathbf{d} + b \quad (6)$$

Now we can express (5) using (6) to give

$$\begin{aligned} \mathbf{d} \bullet\!\!> \mathbf{d}' &\Leftrightarrow \mathbf{w} \cdot \mathbf{d} + b > \mathbf{w} \cdot \mathbf{d}' + b \\ &\Leftrightarrow \mathbf{w} \cdot (\mathbf{d} - \mathbf{d}') > 0 . \end{aligned} \quad (7)$$

Note that the relation $\mathbf{d} \bullet\!\!> \mathbf{d}'$ is expressed in terms of the difference between feature vectors $\mathbf{d} - \mathbf{d}'$, which can be thought of as the combined feature vector of the pair of documents. If we assume that the "true" utility function is indeed linear, the weight vector $\mathbf{w}$ has to satisfy inequality (7) for each pair of documents in the training set. Assuming a finite margin between the n-dimensional feature vectors $\mathbf{d}^{(i)} - \mathbf{d}'^{(i)}$ with $y^{(i)} = +1$ and $y^{(i)} = -1$, we make the constraint (7) stronger and multiply each inequality by $y^{(i)}$,

$$y^{(i)}[\mathbf{w} \cdot (\mathbf{d}^{(i)} - \mathbf{d}'^{(i)})] \geq 1 \qquad i = 1, \ldots, \ell . \quad (8)$$

The weight vector $\mathbf{w}^*$ with optimal generalization is now determined via the principle of structural risk minimization (Vapnik 1995), which – for the present case – leads to the problem of minimizing the squared norm $\|\mathbf{w}\|^2$ of the weight vector under the constraints (8).

According to the Support Vector training algorithm (Cortes & Vapnik 1995), we arrive at the problem of maximizing

$$\begin{aligned} L(\boldsymbol{\alpha}) &= \sum_{i=1}^{\ell} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{\ell} \alpha_i \alpha_j y^{(i)} y^{(j)} \\ &\times \quad [(\mathbf{d}^{(i)} - \mathbf{d}'^{(i)}) \cdot (\mathbf{d}^{(j)} - \mathbf{d}'^{(j)})] , \end{aligned} \quad (9)$$

w.r.t. the α_i. This constitutes a standard quadratic programming problem. Also note that due to the expansion of the last term in (9), the solution $\boldsymbol{\alpha}^*$ to this problem can be calculated solely in terms of the inner products between document descriptions without reference to the descriptions themselves. This fact will be exploited in the following section for the generalization of the method to nonlinear utility functions. Moreover, the optimal weight vector $\mathbf{w}^*$ can be written as a linear combination of differences of document vectors from the training set:

$$\mathbf{w}^* = \sum_{i=1}^{\ell} \alpha_i y^{(i)} (\mathbf{d}^{(i)} - \mathbf{d}'^{(i)}) . \quad (10)$$

All those pairs of documents with $\alpha_i^* \neq 0$ "support" the construction of the optimal hyperplane in the space of

document pairs, and are therefore referred to as "support vectors" (Cortes & Vapnik 1995). Usually, the number of support vectors $\ell_{SV} \ll \ell$, and it is this sparseness that makes the representation (10) so appealing.

After learning, the utility function is represented by the vector $\boldsymbol{\alpha}^*$ together with the training set S. A new pair of documents $(\mathbf{d}, \mathbf{d}')$ is then classified – using (7) and (10) – according to

$$\mathbf{d} \bullet\!\!> \mathbf{d}' \Leftrightarrow \sum_{i=1}^{\ell} \alpha_i^* y^{(i)} [(\mathbf{d}^{(i)} - \mathbf{d}'^{(i)}) \cdot (\mathbf{d} - \mathbf{d}')] > 0 \,. \quad (11)$$

However, combining (6) and (10) it is also possible to reconstruct the utility function of a document $\mathbf{d}$ as

$$U(\mathbf{d}) = \sum_{i=1}^{\ell} \alpha_i^* y^{(i)} (\mathbf{d}^{(i)} - \mathbf{d}'^{(i)}) \cdot \mathbf{d} \,. \quad (12)$$

Both these calculations – equations (11) and (12) – benefit from the sparseness of the expansion (10), which significantly reduces their computational costs.

Extension to the Nonlinear Case

Equation (9) as a direct derivation of (5) assumes a linear model of the utility function $U(\mathbf{d})$. In order to extend the model to include utility functions $U(\mathbf{d})$, which are nonlinear in the features d_k, we define a mapping $\phi : \mathcal{D} \mapsto \mathcal{F}$ from the space $\mathcal{D}$ to an m-dimensional space $\mathcal{F}$, where the dimensionality of $\mathcal{F}$ may be much greater than that of $\mathcal{D}$, $m \gg n$. If we now adopt our linear model in the space $\mathcal{F}$, we obtain

$$U(\mathbf{d}) \quad = \tilde{\mathbf{w}} \cdot \phi(\mathbf{d}) \,. \quad (13)$$

Note, that $\tilde{\mathbf{w}} \in \mathcal{F}$, which means that m nonlinear features can now be taken into account. Using the shorthand notation $(\phi(\mathbf{d}^{(i)}) - \phi(\mathbf{d}'^{(i)})) = \Delta_{\phi^{(i)}}$ and (13), equation (9) becomes

$$L(\boldsymbol{\alpha}) = \sum_{i=1}^{\ell} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{\ell} \alpha_i \alpha_j y^{(i)} y^{(j)} \Delta_{\phi^{(i)}} \cdot \Delta_{\phi^{(j)}} \quad (14)$$

which has to maximized w.r.t. to the α_i.

Our derivation again results in a functional, which only depends on the inner products between document vectors, this time calculated in $\mathcal{F}$. According to the Hilbert–Schmidt theory, for a given space $\mathcal{F}$ there exists a function $K : \mathcal{D} \times \mathcal{D} \mapsto \Re$ – the "kernel function" – that corresponds to an inner product in $\mathcal{F}$. Conversely, we can fix the kernel function $K(\mathbf{d}, \mathbf{d}')$,

$$K(\mathbf{d}, \mathbf{d}') \quad = \quad \phi(\mathbf{d}) \cdot \phi(\mathbf{d}') \,, \quad (15)$$

which corresponds to taking the inner product in some space $\mathcal{F}$ under conditions given by Mercer's theorem (Aizerman, Braverman, & Rozonoer 1964).

We can apply this "kernel trick" to expression (14) which makes it possible to efficiently calculate the dot products in $\mathcal{F}$ for equation (14) by simply evaluating the corresponding kernel function in $\mathcal{D}$. Similiar arguments apply to the evaluation of the equations for classification (11) and the computation of the utility function (12) in the nonlinear case.

As an example for an admissible kernel function consider

$$K(\mathbf{d}, \mathbf{d}') = (\mathbf{d} \cdot \mathbf{d}' + 1)^p \,, \quad (16)$$

which corresponds to the space $\mathcal{F}$ of all monomials of the n input features up to degree p (Vapnik 1995). For document descriptions this corresponds to taking into account higher order correlations of word occurrances. In particular for binary document descriptions indicating the occurrance of particular keywords, a polynomial utility function can be interpreted as a weighted logical expression in the sense of a query. The most important advantage of the kernel technique is the enormous reduction in computational costs as opposed to explicitly performing the mapping ϕ and then taking the dot product in $\mathcal{F}$. For $p = 2$ and $n \geq 10\,000$ (not uncommon in IR) in (16) the dimensionality m of the corresponding feature space $\mathcal{F}$ is $m \geq 50\,015\,000$ (Burges 1997). If we did not use the "kernel trick", we would have to transform the documents to a ≈ 50 million dimensional space in order to compute the inner products.

Experimental Results

100 data points $\mathbf{d}$ were generated from a uniform distribution over the unit square $[0,1] \times [0,1]$. 10 points were used to generate the training set, 90 were set aside for the test set. A utility value $U(\mathbf{d})$ was assigned to each data point with (a) a linear function $U(\mathbf{d}) = d_1 + 2d_2$ and (b) a quadratic function $U(\mathbf{d}) = d_1 + 2d_2 - 4d_1 d_2$. All document pairs of the training set were labeled according to (1) and (5). We used the kernel given in (16), which should be capable of modeling polynomial utility functions. The algorithm was trained using a modification of Steve Gunn's Support Vector implementation in MATLAB. Training was done for values $p = 1 \ldots 5$, and we determined the degree p of the optimal kernel by minimizing an upper bound on the generalization error given by (Cortes 1995)

$$\left[\max_{j=1,\ldots,\ell} \left\| \Delta_{\phi^{(j)}} - \sum_{i=1}^{\ell} \Delta_{\phi^{(i)}} \right\|^2 \right] \|\tilde{\mathbf{w}}^*\|^2 \,, \quad (17)$$

which can be evaluated conveniently using the "kernel trick". The results are depicted in Figure 1, (a) and (b), for the linear and quadratic utility function, respectively. From the iso-utility lines it can be seen that in both cases the utility function found by the algorithm is very similar to the one used to generate the data. Indicated by diamonds are document vectors that were part of "support vector pairs", whose numbers are given in the plot. Note how "support vector pairs" are close in the sense that their utilities are similar. Since only for these pairs $\alpha_i^* \neq 0$, they uniquely determine the utility function. To obtain a test error we calculated the percentage of misclassified document pairs from the test

set. This error was 0.30% for the linear case and 2.2% for the quadratic utility function.

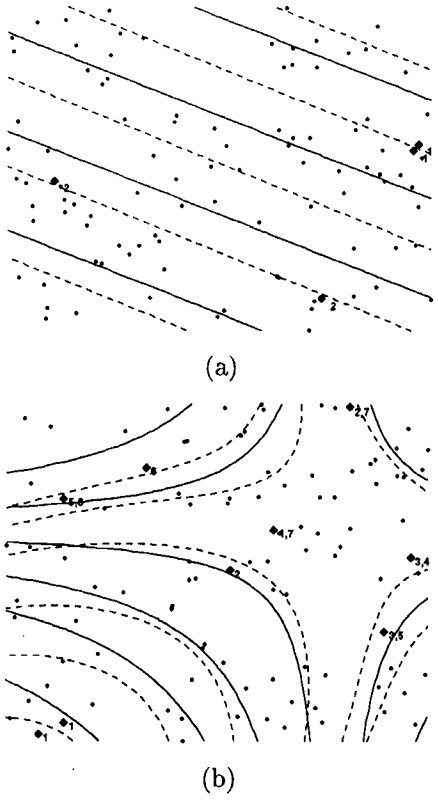

Figure 1: Contour plot representation of the original and the reconstructed utility functions. Solid lines indicate iso-utility for the true utility function, dashed lines show the iso-utility for the utility function recovered by the algorithm for **(a)** a linear utility function $U(\mathbf{d}) = d_1 + 2d_2$ and **(b)** a quadratic utility function $U(\mathbf{d}) = d_1 + 2d_2 - 4d_1d_2$. The degree automatically chosen by the algorithm was $p = 1$ in the linear and $p = 3$ in the quadratic case.

Discussion

In this paper, we investigate the problem of learning a preference relation from a given set of document pairs $(\mathbf{d}, \mathbf{d}')$, an approach which is based on ordinal utilities, by learning a mapping from documents to utilities. This approach is also related to Robertson's "probability ranking principle" (Robertson 1977):

$$\mathbf{d} \diamond\!\!\succ \mathbf{d}' \quad \Leftrightarrow \quad P(R|\mathbf{d}) > P(R|\mathbf{d}') \qquad (18)$$

where $P(R|\mathbf{d})$ is interpreted as probability of *usefulness* of $\mathbf{d}$. If we assign utility values $U(\mathbf{d})$ to documents via the strictly monotonically increasing transformation $U(\mathbf{d}) = \ln \frac{P(R|\mathbf{d})}{1-P(R|\mathbf{d})}$, a linear utility function is obtained if the individual features are independent w.r.t. R and its complement. Transforming back to $P(R|\mathbf{d})$ we obtain

$$P(R|\mathbf{d}) \quad = \quad \frac{1}{1 + \exp(-U(\mathbf{d}))}. \qquad (19)$$

In analogy to this we can interpret $1/(1+\exp(-U(\mathbf{d})))$ as probability of *usefulness* of $\mathbf{d}$ for a nonlinear utility function $U(\mathbf{d})$ without making the assumption of probabilistic independence of features.

Acknowledgments

This project was funded by the Technical University of Berlin via the Forschungsinitiativprojekt FIP 13/41.

References

Aizerman, M.; Braverman, E.; and Rozonoer, L. 1964. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control* 25:821–837.

Bishop, C. M. 1995. *Neural Networks for Pattern Recognition.* Oxford: Clarendon Press.

Bollmann, P., and Wong, S. K. M. 1987. Adaptive linear information retrieval models. In *Proceedings of the 10th Annual ACM SIGIR Conference an Research and Development in Information Retrieval*, 157–163.

Burges, C. J. 1997. A tutorial on Support Vector Machines for pattern recognition. submitted to Data Mining and Knowldge Discovery.

Cortes, C., and Vapnik, V. 1995. Support Vector Networks. *Machine Learning* 20:273–297.

Cortes, C. 1995. *Prediction of Generalization Ability in Learning Machines.* Ph.D. Dissertation, University of Rochester, Rochester, USA.

Joachims, T. 1997. Text categorization with Support Vector Machines: Learning with many relevant features. Technical report, University Dortmund, Department of Artifical Intelligence. LS–8 Report 23.

Roberts, F. 1979. *Measurement Theory.* Massachusetts: Addison Wesley.

Robertson, S. 1977. The probability ranking principle in IR. *Journal of Documentation* 33(4):294–304.

Salton, G. 1968. *Automatic Information Organization and Retrieval.* New York: McGraw–Hill.

Salton, G. 1989. *Automatic Text Processing.* Massachusetts: Addison Wesley.

Suppes, P.; Krantz, D. H.; Luce, R. D.; and Tversky, A. 1989. *Foundations of Measurement Vol. II.* San Diego: Academic Press Inc.

Vapnik, V. 1995. *The Nature of Statistical Learning Theory.* New York: Springer–Verlag.

Wong, S. K. M.; Yao, Y. Y.; and Bollmann, P. 1988. Linear structure in information retrieval. In *Proceedings of the 11th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 219–232.