Case-based Label Ranking

Klaus Brinker[1] and Eyke Hüllermeier[1]

Data and Knowledge Engineering,
Otto-von-Guericke-Universität Magdeburg, Germany
{brinker,huellerm}@iti.cs.uni-magdeburg.de

Abstract. Label ranking studies the problem of learning a mapping from instances to rankings over a predefined set of labels. We approach this setting from a case-based perspective and propose a sophisticated k-NN framework as an alternative to previous binary decomposition techniques. It exhibits the appealing property of transparency and is based on an aggregation model which allows to incorporate a broad class of pairwise loss functions on label ranking. In addition to these conceptual advantages, we also present empirical results underscoring the merits of our approach in comparison to state-of-the-art learning methods.

1 Introduction

The topic of learning preferences has attracted increasing attention recently and contributes to the more general trend of investigating complex and structured output spaces in machine learning, such as label sequences or natural language parsing trees [1, 2]. Label ranking, a particular preference learning scenario, studies the problem of learning a mapping from instances to rankings over a finite number of predefined *labels* (alternatives). It can be considered as a natural generalization of conventional classification, where only a single label (the top-label) is requested instead of a ranking of all labels. Applications of label ranking can be found in various fields such as, e.g., natural language processing and text categorization. Among those approaches proposed to address this category of learning problems the following techniques provide a general means to extend arbitrary (linear) binary classification algorithms: *Ranking by pairwise comparison* as a natural extension of pairwise classification [6] and *constraint classification* which aims at learning a linear utility function for each label [7].

Case-based learning algorithms have been applied successfully in various fields such as, e.g., machine learning and pattern recognition [8]. These algorithms defer processing the training data until an estimation for a new instance is requested, a property distinguishing this class of learning methods from typical model-based approaches. As a particular advantage of delayed processing, these learning methods may estimate the target function locally instead of inducing a global prediction model for the entire input domain from the data. Predictions are typically obtained using only a small, locally restricted subset of the entire training data, therefore supporting a human-accessible estimation process for target objects. This appealing property is difficult to realize in algorithms using

complex global models of the target function and critical to certain applications where black-box predictions are not acceptable. In fact, this drawback arises already in binary classification, however, its severity increases in label ranking as a consequence of the more complex structure of the underlying target space: Previous approaches on extending binary classification, such as the above-mentioned techniques, generate inflated training sets embedded in a higher dimensional feature space to encode preference constraints, or perform a decomposition into multiple binary classification tasks. Hence, both techniques increase the complexity level of constructing a prediction model such that it becomes difficult if not impossible to provide a comprehensible prediction explanation.

Besides the question of transparency in target prediction, the process of decomposing label rankings into binary preferences is accompanied by a loss of information as from these binary training inputs alone, the original rankings cannot be recovered [9]. As a result, it implicitly restricts the class of loss functions which can potentially be minimized. In this paper, we propose a case-based approach to label ranking as an alternative which maintains the appealing property of transparency and is based on an aggregation model which allows to incorporate a broad class of loss functions. In contrast to previous approaches, it also provides a natural means of calculating confidence scores for predictions.

The subsequent section introduces the basic problem of label ranking and a general architecture to approach this problem in a case-based learning framework. Section 3 studies the subcomponent of aggregating label rankings with respect to common choices of pairwise loss functions. Model-based approaches to label ranking are reviewed in Section 4. Section 5 is devoted to an empirical evaluation of case-based label ranking and a comparison with the pairwise ranking technique. Finally, various extensions and modifications of the basic approach are discussed in Section 6.

2 Framework

In label ranking, the problem is to learn a mapping from instances x (e.g., representing a person) of an instance space $\mathcal{X}$ to rankings $\succ_x$ (total strict orders) over a finite set of labels $\mathcal{L} = \{\lambda_1, \ldots, \lambda_c\}$, where $\lambda_i \succ_x \lambda_j$ means that instance x prefers label λ_i to λ_j. A ranking over $\mathcal{L}$ can be represented by a permutation as there exists a unique permutation τ such that $\lambda_i \succ_x \lambda_j$ iff $\tau(i) < \tau(j)$, where $\tau(i)$ denotes the position of the label λ_i in the ranking. The target space of all permutations over c labels will subsequently be referred to as $\mathcal{S}_c$. Let us make the idealized assumption that the training data submitted to the learning algorithm consists of a set of instances $(x_1, \tau_1), \ldots, (x_m, \tau_m)$ which contain the *complete* label rankings and therefore the entire sets of pairwise preferences. Of course, in practice it might not always be possible to observe complete rankings. However, by reducing the technical complexity, this assumption will allow us to focus on the main components of case-based label ranking. Later on, we will discuss how to handle the more general case where only a subset of all pairwise preferences are available for each training example x_i.

In the following, we will discuss a general case-based framework for learning label rankings. The k-nearest neighbor algorithm (k-NN) is arguably the most basic case-based learning method [5]. In its simplest version, it assumes all instances to be represented by feature vectors $x = ([x]_1, \ldots, [x]_N)^\top$ in the N-dimensional space $\mathcal{X} = \mathbb{R}^N$ endowed with the standard Euclidian metric as a distance measure, though an extension to other instance spaces and more general distance measures $d(\cdot, \cdot)$ is straightforward. When a query feature vector x is submitted to the k-NN algorithm, it computes the k training instances closest to this point in terms of $d(\cdot, \cdot)$. In the case of classification learning, the k-NN algorithm estimates the associated class label by the most frequent label among those training instances. It can be adapted to the regression learning scenario by replacing the majority voting step with computing the (weighted) mean of the target values.

A unified view of both classification and label ranking as discrete-valued learning problems provides a straightforward generalization of the k-NN algorithm such that the most common label ranking is used as the predicted target object. However, several drawbacks make this approach seem inappropriate in general:

- The cardinality of the target space in label ranking is $|\mathcal{S}_c| = c!$, a number far beyond the typical cardinality in classification learning. Therefore, if the local distribution of label rankings does not have sharp peaks, equal votes statistics are much more likely (except for $k = 1$). Random tie-breaking, a standard technique in k-NN learning, may be used rather frequently resulting in randomly selecting a label ranking among the k nearest neighbors.
- In contrast to classification learning, where only the discrete metric (0/1 loss) is given on the target space, *non-trivial* metrics can be defined on label rankings, a property shared with regression learning. The discrete k-NN algorithm does not exploit this property in the aggregation step.

To avoid the above-mentioned drawbacks, a more sophisticated algorithm should incorporate the structured nature of the space of label rankings. Our approach considers aggregation techniques for label ranking which are conceptually related to averaging in k-NN regression learning. To this end, we incorporate a common rank aggregation model in order to combine the k nearest neighbors into a single ranking. This model has been used in a variety of applications, such as in combining meta-search results, however, it is a novel component in a label ranking algorithm. The *consensus label ranking* is computed such that it minimizes the sum of pairwise disagreement indices with respect to all k rankings. The corresponding formal model will be detailed in Section 3.

3 Aggregating Label Rankings

The problem of aggregating rankings, i.e., to merge a finite set of rankings into a single consensus ranking in an suitable manner, arises in a variety of applications. Among those are retrieval-related database applications, combining experts and

the combination of meta-search results [10]. Moreover, the range of scientific disciplines studying aggregation techniques extends to social choice theory where the problem of combining votes constitutes a well-studied research topic.

In order to analyze the problem of aggregating label rankings in a formal manner, let $\tau_1, \ldots, \tau_k$ denote rankings of the c alternatives $\lambda_1, \ldots, \lambda_c$. A common method to measure the quality of a ranking τ as an aggregation of the set of rankings $\tau_1, \ldots, \tau_k$ is to compute the sum of pairwise loss values

$$L(\tau) \stackrel{\text{def}}{=} \sum_{i=1}^{k} l(\tau, \tau_i)$$

with respect to a loss function $l : \mathcal{S}_c \times \mathcal{S}_c \to \mathbb{R}_{\geq 0}$ defined on pairs of rankings.

Having specified a loss function $l(\cdot)$, this model leads to the optimization problem of computing a ranking $\hat{\tau} \in \mathcal{S}_c$ (not necessarily unique) such that

$$L(\hat{\tau}) = \min_{\tau \in \mathcal{S}_c} \sum_{i=1}^{k} l(\tau, \tau_i). \tag{1}$$

Common choices for the loss function are the *Kendall tau loss* [11], the sum of absolute rank distances, which is also referred to as *Spearman footrule loss* [10], and the sum of squared rank distances. The linear transformation of the latter loss function into a similarity measure is well-known as the *Spearman rank correlation coefficient* [12, 13]. The Kendall tau loss l_K essentially calculates the number of pairwise rank inversions on labels to measure the ordinal correlation of two rankings. More formally,

$$l_K(\tau, \tau') \stackrel{\text{def}}{=} \big|\{(i,j) \mid \tau(i) < \tau(j) \wedge \tau'(i) > \tau'(j)\}\big|. \tag{2}$$

The Spearman footrule loss l_1 and the sum of squared rank distances loss l_2 are formally defined as

$$l_1(\tau, \tau') \stackrel{\text{def}}{=} \sum_{i=1}^{c} |\tau(i) - \tau'(i)| \quad \text{and} \quad l_2(\tau, \tau') \stackrel{\text{def}}{=} \sum_{i=1}^{c} (\tau(i) - \tau'(i))^2. \tag{3}$$

In the following, we will elaborate on solving the optimization problem (1) depending on the particular choice of the loss function. When using the Kendall tau loss, the associated optimal solution is also referred to as the *Kemeny-optimal ranking*. Kendall's tau is an intuitively quite appealing loss function and Kemeny-optimal rankings have several nice properties. Among those, they satisfy the so-called *Condorcet criterion*, which states that if a certain label defeats every other label in pairwise majority voting among the rankings, this label should be ranked first. However, it has been shown in [14] that the problem of computing Kemeny-optimal rankings is NP-hard.

In the case of the Spearman footrule loss, the optimization problem (1) is equivalent to finding a minimum cost maximum matching in a bipartite graph

with c nodes [15]. Fagin et al. [10] proposed a computationally efficient approximate aggregation algorithm which for complete rankings simplifies to ordering the labels according to their median ranks, a task which can be accomplished in $\mathcal{O}(kc + c \log c)$ time. In terms of accuracy, Fagin et al. [10] proved that this algorithm computes a constant-factor approximation $\bar{\tau}$ of the optimal solution for both the l_1 and the l_K loss function. More precisely,

$$\min_{\tau \in \mathcal{S}_c} \sum_{i=1}^{k} l_1(\tau, \tau_i) \leq 2 \sum_{i=1}^{k} l_1(\bar{\tau}, \tau_i) \quad \text{and} \quad \min_{\tau \in \mathcal{S}_c} \sum_{i=1}^{k} l_K(\tau, \tau_i) \leq 4 \sum_{i=1}^{k} l_K(\bar{\tau}, \tau_i). \quad (4)$$

Moreover, Dwork et al. [15] showed that median ordering indeed finds the optimal solution for the l_1 loss in the case of *unique* median ranks. In the case of equal median ranks, we shall apply random tie breaking.

For the sum of squared rank distances loss, a provably optimal solution of (1) is obtained by ordering alternatives according to the so-called *Borda count* [9], a voting technique well-known in social choice theory. The Borda count of an alternative is the number of (weighted) votes for that alternative in pairwise comparisons with all remaining options. This voting rule requires computational time in the order of $\mathcal{O}(kc + c \log c)$ and thus can be evaluated very efficiently.

In the experimental section, the Borda-count and the median ordering techniques will be incorporated into the learning algorithm as they are computationally efficient and have a sound theoretical basis. However, as the aggregation component is an isolated module within our case-based framework, alternative aggregation techniques which may be suitable for the particular application at hand may be integrated naturally (such as aggregation techniques which minimize loss functions focusing on correct top ranks rather than distributing equal weights to all positions).

As an appealing property of this aggregation model, average loss values, $\frac{1}{k} \sum_{i=1}^{k} l(\tau, \tau_i)$, provide a natural means of associating a (reversed) confidence score with a prediction τ, in contrast to previous approaches (cf. section 4) where techniques for calculating confidence scores have not been proposed yet. Moreover, it is convenient to rescale to the unit interval by

$$1 - \frac{1}{k} \sum_{i=1}^{k} \frac{l(\tau, \tau_i)}{\max_{\hat{\tau}, \hat{\tau}' \in \mathcal{S}_c} l(\hat{\tau}, \hat{\tau}')} \quad (5)$$

such that higher scores correspond to more reliable predictions. We will provide empirical evidence in the experimental section that this approach indeed yields a meaningful measure of confidence. Complementing the appealing property of an accessible model, case-based ranking supports critical applications where a transparent and reliable prediction process is a mandatory requirement.

4 Model-based Approaches

One natural approach for modeling preference rankings is to represent each individual label by means of an associated (real-valued) utility function. More

precisely, a utility function $f_i : \mathcal{X} \to \mathbb{R}$ is used for modeling each of the labels λ_i, $i = 1 \ldots c$, where $f_i(x)$ is the utility assigned to λ_i by the instance x. To obtain a ranking for x, the labels are ordered according to these utility scores, i.e., $\lambda_i \succeq_x \lambda_j \Leftrightarrow f_i(x) \geq f_j(x)$.

If the training data would offer the utility scores directly, preference learning in the label ranking scenario would reduce to a standard regression problem. This information can rarely be assumed available, however. Instead, usually only constraints derived from comparative preference information of the form "This label should have a higher utility score than that label" are given. Thus, the challenge for the learner is to find a function that is as much as possible in agreement with all constraints. A corresponding method for learning *linear* utility functions $f_i(\cdot)$, $i = 1 \ldots c$, from training data has been proposed in the framework of *constraint classification* [7]. Constraint classification encodes each constraint induced by comparative preference information as a binary training example. The utility functions can be learned by solving the overall binary classification problem by means of an arbitrary linear learning algorithm.

An alternative pairwise approach to label ranking learning has been introduced in [6]. The key idea in pairwise label ranking is to learn, for each pair of labels (λ_i, λ_j), $i < j$, a binary predicate $\mathcal{M}_{ij}(x)$ that predicts whether $\lambda_i \succ_x \lambda_j$ or $\lambda_j \succ_x \lambda_i$ for an input x. In order to rank the labels for a new instance, predictions for all pairwise label preferences are obtained and a ranking that is maximally consistent with these preferences is derived, typically by means of a voting scheme. This approach is a natural extension of pairwise classification, i.e., the idea to tackle a multi-class classification problem by learning separate theories for each pair of classes.

5 Empirical Evaluation

The purpose of this section is to provide an empirical evaluation of case-based label ranking in terms of accuracy and computational complexity. The first series of experiments has been set up in order to compare case-based label ranking (k-NN-LR) using Borda and median aggregation with the pairwise label ranking framework [6], where support vector machines [16] with linear (PW-SVM-LIN) and RBF kernels (PW-SVM-RBF) were used as the binary base learner. For pairwise ranking, we also considered the common Borda technique for aggregating comparative pairwise preference votes received from the binary classification models. The constraint classification approach has not been included in the experimental evaluation as earlier experiments suggested that it typically achieves a level of accuracy comparable to pairwise label ranking while being computationally far more demanding in general [3].

As benchmark datasets of sufficient size are not publicly available for label ranking, we simulated this setting using the following multi-class datasets from the UCI Repository of machine learning databases [17] and the Statlog collection [18]: Iris, Wine, Glass, Vowel, Vehicle. For each of these datasets, we

	Iris	Wine	Glass	Vowel	Vehicle
PW-SVM-LIN	0.767 ±0.148	0.910 ±0.088	0.827 ±0.054	0.484 ±0.040	0.788 ±0.040
PW-SVM-RBF	0.967 ±0.047	0.905 ±0.083	0.842 ±0.058	0.864 ±0.021	0.857 ±0.027
KNN-LR-MEDIAN	0.940 ±0.066	0.927 ±0.045	0.831 ±0.091	0.864 ±0.023	0.795 ±0.039
KNN-LR-BORDA	0.940 ±0.058	0.933 ±0.051	0.831 ±0.091	0.864 ±0.023	0.793 ±0.051
PW-SVM-LIN	0.867 ±0.074	0.941 ±0.055	0.923 ±0.039	0.769 ±0.028	0.892 ±0.027
PW-SVM-RBF	0.980 ±0.023	0.958 ±0.056	0.929 ±0.037	0.962 ±0.009	0.909 ±0.019
KNN-LR-MEDIAN	0.963 ±0.029	0.949 ±0.039	0.898 ±0.085	0.957 ±0.008	0.876 ±0.037
KNN-LR-BORDA	0.967 ±0.031	0.969 ±0.024	0.892 ±0.080	0.957 ±0.008	0.887 ±0.029
PW-SVM-LIN	0.844 ±0.082	0.933 ±0.063	0.891 ±0.044	0.673 ±0.029	0.861 ±0.024
PW-SVM-RBF	0.978 ±0.031	0.944 ±0.053	0.899 ±0.044	0.922 ±0.014	0.896 ±0.017
KNN-LR-MEDIAN	0.960 ±0.044	0.937 ±0.052	0.882 ±0.075	0.922 ±0.013	0.854 ±0.032
KNN-LR-BORDA	0.960 ±0.044	0.952 ±0.039	0.882 ±0.075	0.922 ±0.013	0.853 ±0.038

Table 1. Empirical comparison of case-based label ranking (KNN-LR) using Borda and median aggregation with the model-based pairwise ranking approach, where support vector machines with linear (PW-SVM-LIN) and RBF kernels (PW-SVM-RBF) were used as the binary base learner. The empirical results are grouped in three separate sections, where the Spearman footrole (first section), the Spearman rank correlation (second section) and the Kendall tau (third section) evaluation measures were computed on the testsets (and used in the respective experiments for tuning hyperparameters on the training sets).

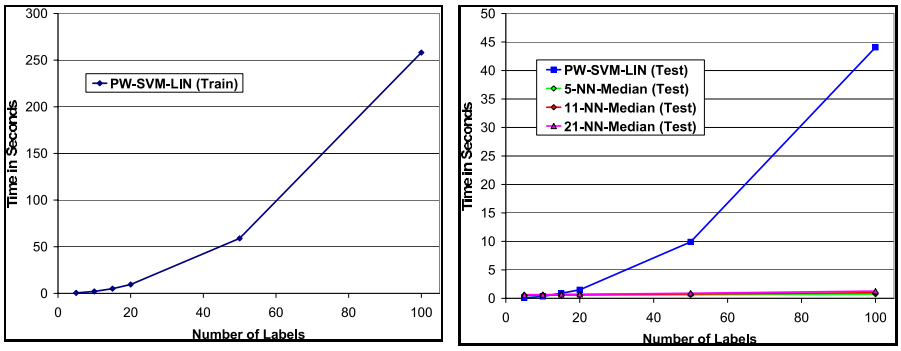
trained a naive Bayes classifier[1] and then for each instance, *all* the labels present in the respective dataset were ordered with respect to decreasing predicted class probabilities (in the case of ties, labels with lower indices are ranked first).[2]

For linear kernels the margin-error penalty C was chosen from $\{2^{-2}, \ldots, 2^{10}\}$ and for RBF kernels the considered sets of hyperparameters are given by $C \in \{0.5, 1, 5, 10, 50, 100, 1000\}$ and $\gamma \in \{10^{-3}, \ldots, 10^{3}\}$. In the case of k-NN learning, the number of nearest neighbors k was selected from $\{1, 3, \ldots, 15, 21\}$. For all parameters, the optimal values were selected using 10-fold crossvalidation on the training sets where the accuracy was estimated with respect to the metric on label rankings used in the specific experimental run. In order to facilitate interpretability, we employed the Spearman footrule, the squared rank distances and the Kendall tau loss functions (see Section 3) in a common normalized version such that the loss (the similarity value) evaluates to -1 for reversed and to $+1$ for identical label rankings. Hence, for the first set of experiments, the overall experimental setup consists of a nested two level crossvalidation procedure, the inner level for selecting hyperparameters and the outer level for estimating generalization accuracy using an additional crossvalidation step.

As anticipated on behalf of the theoretical results, Borda-aggregation slightly outperforms median-aggregation in the case of the Spearman rank correlation,

[1] We employed the implementation for naive Bayes classification on numerical datasets (`NaiveBayesSimple`) contained in the Weka machine learning package [19].

[2] This setting was previously studied in active learning for label ranking [20].



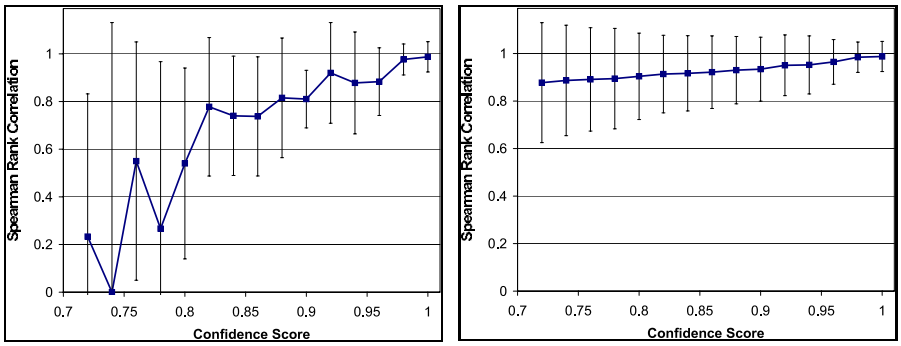
(a) (b)

Fig. 1. Computational Time: (a) Training (results for k-NN-LR are omitted as storing the training data requires only negligible effort). (b) Testing.

while a substantial difference between Borda- and Median-aggregation cannot be observed for the Spearman footrule loss. For Kendall's tau, both aggregation techniques achieve similar results. Surprisingly, our k-NN-LR is competitive with PW-SVM-LIN on three datasets (WINE, GLASS and VEHICLE) and even outperforms it on the remaining two by a large margin (IRIS and VOWEL). PW-SVM-RBF outperforms our k-NN-LR approach on most datasets, however, typically only by a small margin. Moreover, there are several directions for further improving k-NN-LR, such as weighted aggregation, feature selection and similarity learning which have not been incorporated yet (see Section 6).

We conducted a series of controlled experiments in order to study the computational requirements of the proposed case-based framework using a synthetic experimental setting. This setting is a replication of [6] and operates in the context of expected utility theory. In this setting, the optimal pairwise decision boundaries are hyperplanes, therefore, we selected a linear kernel for PW-SVM with $C = 1000$.[3] Setting both the number of training and test instances to 1000, $k \in \{5, 11, 21\}$ and the input dimension to 10, the training and prediction time of PW-SVM-LIN and k-NN-LR-Median (the difference to k-NN-LR-Borda is negligible) was evaluated for $c \in \{5, 10, 15, 20, 50, 100\}$. The experimental results, depicted in Figure 1, demonstrate that even though we implemented k-NN-LR in a non-optimized straightforward version which does not exploit any sophisticated data structures for supporting efficient nearest neighbor search, it performs very well in terms of computational efficiency. The computational complexity of pairwise ranking can be attributed to the fact that the number of binary SVMs to be trained is *quadratic* in the number of labels. In contrast to standard classification learning, the training sets for those binary subproblems have the same

[3] Note that this property prohibits a meaningful comparison between pairwise ranking with a *linear* base learner and k-NN label ranking in terms of accuracy.



(a) (b)

Fig. 2. Accuracy-Confidence Dependence: (a) Average rank correlation values are shown for all predictions associated with a particular confidence level. (b) Rank correlation values have been averaged over all predictions associated with *at least* the shown confidence level.

size as the original dataset, which entails a substantial increase in computational demands. Besides the theoretical difference in testing complexity in this setting ($\mathcal{O}(c^2)$ for pairwise and $\mathcal{O}(kc + c \log c)$ for k-NN ranking), this evaluation underscores the difference in complexity from a practical point of view: k-NN label ranking scales easily to problems with huge numbers of alternatives whereas the computational burden involved with pairwise ranking prohibits its application in this regime in many cases.

An empirical comparison between the simple discrete generalization of k-NN learning (see Section 2) and our approach in the above-described synthetic setting provided strong support for the assumption that majority selection fails to provide a suitable aggregation technique for label rankings: For $c \geq 10$, discrete k-NN always achieves optimal performance for $k = 1$ and is outperformed by k-NN-LR-Median and k-NN-LR-Borda for all considered choices of loss functions by a large margin.

As mentioned in Section 3, the rescaled accumulated loss (5) can be interpreted as a confidence score for the prediction τ. In order to investigate the dependence of the generalization accuracy on the magnitude of the confidence scores, the k-NN-Borda algorithm ($k = 5$) was evaluated in the above-stated setting: The VEHICLE dataset was randomly split into a training and test set of equal size. The predicted label rankings and the associated confidence scores on the test set were used to generate Figure 2, where in (a) the average Spearman rank correlation was averaged over all predictions associated with a *particular* discrete confidence level whereas in (b) the rank correlation was averaged over all predictions with *at least* the specified confidence level. The confidence-accuracy curves clearly indicate that indeed the proposed confidence measure is strongly

correlated with the accuracy of predictions. Hence, rejection strategies which refuse to make a prediction if the associated confidence level is below a certain threshold may further increase accuracy, e.g., if we predicted a label ranking for a test instance only if the associated confidence score equals 1.0 (which covers roughly 34% of the entire test set), the Spearman rank correlation would increase from the base level of 0.863 to 0.988 on this subset! Rejecting only the 10% least confident predictions already increases the remaining average rank correlation to 0.913. Qualitatively similar observations can be made on the remaining datasets whenever $k > 1$.

6 Extensions

Our general framework for case-based label ranking can be naturally extended to accommodate for the specific kind of preference information available, such as to the scenarios where a total order is only specified for a subset of all labels (*partial rankings*), ties among labels are permitted, or only certain pairwise preferences can be observed (such as for typical text categorization settings). Generalized loss functions, such as the extensions to Kendall's tau and the Spearman footrule loss defined in [10], can be plugged into the aggregation model (1). A particularly interesting generalization of label ranking is *calibrated label ranking* as introduced in [4]. Roughly speaking, a calibrated ranking is a ranking with an additional neutral label which splits a ranking into two parts, thus combining label ranking and multilabel classification learning. We applied our case-based approach in this setting, using the `Reuters-2000` text data as a benchmark, and obtained very promising results (omitted here due to reasons of space). Even though these results are preliminary, they indicate that our approach is indeed amenable to more general preference learning scenarios.

Ha and Haddawy [21] proposed an appealing *probabilistic loss* on preferences which originates from the Kendall tau loss and extends to both *partial* and *uncertain* preferences. Efficient methods for (approximate) rank aggregation with respect to this measure have not been developed yet but could potentially be plugged into the case-based label ranking framework in order to generalize to the uncertainty case. Moreover, Ha and Haddawy [21] developed a successful case-based system for collaborative filtering, where a new user is matched to the user with the most similar preference structure in the database to provide movie recommendations. This system computes similarity between objects (users) with respect to partial preferences as opposed to label ranking where *separate* features are assumed to be available.

Another interesting direction for extending the basic k-NN label ranking algorithm is to consider a weighted aggregation model where the loss values are weighted by the distances α_i of the respective feature vectors to the query vector, $L'(\hat{\tau}) = \min_{\tau \in \mathcal{S}_c} \sum_{i=1}^{k} \alpha_i \, l(\tau, \tau_i)$. Chin et al. [22] studied a weighted variant of the Kendall tau loss function and proposed a polynomial time approximate aggregation algorithm.

The computational complexity involved in searching the k nearest neighbors can be reduced by using sophisticated data structures, such as metric and kd-trees, and in particular for high dimensional problems, there exist efficient techniques for computing *approximate nearest neighbors* [23]. More precisely, these techniques may be used to improve scaling complexity in terms of the number of instances.

A well-known drawback of the k-NN family of algorithms is its sensitivity to the similarity metric used to determine distances. Besides feature selection methods, a variety of sophisticated algorithms have been proposed which aim at *learning* an appropriate similarity measure based on the given training set in order to increase robustness and boost accuracy [24]. Integrating similarity learning into k-NN label ranking seems to be another promising direction of further research.

Regarding theoretical foundations, it would be interesting to transfer existing results on the performance of k-NN estimation (asymptotically valid bounds on the estimation error [5]) to the label ranking scenario. Preliminary investigations indicate that this is indeed possible, though a detailed discussion is clearly beyond the scope of this paper.

7 Conclusion

Despite its conceptual simplicity, case-based learning is one of the most efficient approaches to conventional machine learning problems like classification and possesses a number of appealing properties. The case-based approach thus lends itself to be applied in label ranking, a recently introduced more complex type of learning problem. The results in this paper show that case-based label ranking indeed provides a viable alternative to model-based approaches. Beyond the conceptual benefits of flexibility in terms of pairwise loss functions, transparency and confidence computation, the empirical evaluation demonstrates that k-NN label ranking achieves results comparable to state-of-the-art model-based approaches while being amenable to the regime of large-scale problems. Generalizing binary classification techniques to label ranking learning in a model-based methodology suffers substantially from the increased complexity of the target space in ranking (in comparison to classification or regression learning), thus, yielding high computational complexity even for moderately complex problems. This contrasts with the k-NN approach, where the complexity of the target space solely affects the aggregation step and, hence, carries much less weight.

Acknowledgments
This research was supported by the German Research Foundation (DFG) and Siemens Corporate Research (Princeton, USA).

References

1. Tsochantaridis, I., Hofmann, T., Joachims, T., Altun, Y.: Support vector machine learning for interdependent and structured output spaces. In: ICML '04: Proceed-

ings of the twenty-first international conference on Machine learning, New York, NY, USA, ACM Press (2004) 823–830
2. Altun, Y., McAllester, D., Belkin, M.: Margin semi-supervised learning for structured variables. In Weiss, Y., Schölkopf, B., Platt, J., eds.: Advances in Neural Information Processing Systems 18. MIT Press, Cambridge, MA (2006)
3. Brinker, K., Fürnkranz, J., Hüllermeier, E.: Label Ranking by Learning Pairwise Preferences. Submitted.
4. Brinker, K., Fürnkranz, J., Hüllermeier, E.: A unified model for multilabel classification and ranking. Proc. ECAI-2006. To appear.
5. Dasarathy, B.V.: Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques. IEEE Computer Society Press, Los Alamitos, California (1991).
6. Fürnkranz, J., Hüllermeier, E.: Pairwise preference learning and ranking. In: Proceedings of the 14th European Conference on Machine Learning (ECML 2003), Cavtat, Croatia, Springer-Verlag (2003) 145–156
7. Har-Peled, S., Roth, D., Zimak, D.: Constraint classification: A new approach to multiclass classification and ranking. In: Advances in Neural Information Processing Systems 15 (NIPS 2002). (2002)
8. Aha, D.W., Kibler, D., Alber, M.K.: Instance-based learning algorithms. Machine Learning **6**(1) (1991) 37–66
9. Hüllermeier, E., Fürnkranz, J.: Ranking by pairwise comparison: A note on risk minimization. In: IEEE International Conference on Fuzzy Systems (FUZZ-IEEE–04). (2004)
10. Fagin, R., Kumar, R., Mahdian, M., Sivakumar, D., Vee, E.: Comparing and aggregating rankings with ties. In: Proc. 23rd ACM Symposium on Principles of Database Systems (PODS). (2004) 47–58
11. Kendall, M.G.: Rank correlation methods. Charles Griffin, London (1955)
12. Spearman, C.: The proof and measurement of association between two things. American Journal of Psychology **15** (1904) 72–101
13. Lehmann, E.L., D'Abrera, H.J.M.: Nonparametrics: Statistical Methods Based on Ranks, rev. ed. Prentice-Hall, Englewood Cliffs, NJ (1998)
14. Bartholdi, J.J., Tovey, C.A., Trick, M.A.: Voting schemes for which it can be difficult to tell who won the election. Social Choice and welfare **6**(2) (1989) 157–165
15. Dwork, C., Kumar, R., Naor, M., Sivakumar, D.: Rank aggregation methods for the web. In: World Wide Web. (2001) 613–622
16. Schölkopf, B., Smola, A.J.: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press, Cambridge, MA (2002)
17. Blake, C.L., Merz, C.J.: UCI repository of machine learning databases (1998) Data available at `http://www.ics.uci.edu/~mlearn/MLRepository.html`.
18. Michie, D., Spiegelhalter, D.J., Taylor, C.C.: Machine Learning, Neural and Statistical Classification. Ellis Horwood (1994) Data available at `ftp.ncc.up.pt/pub/statlog/`.
19. Witten, I.H., Frank, E.: Data Mining: Practical machine learning tools with Java implementations. Morgan Kaufmann, San Francisco (2000)
20. Brinker, K.: Active learning of label ranking functions. In Greiner, R., Schuurmans, D., eds.: Proceedings of the Twenty-First International Conference on Machine Learning (ICML 2004). (2004) 129–136
21. Ha, V., Haddawy, P.: Similarity of personal preferences: theoretical foundations and empirical analysis. Artif. Intell. **146**(2) (2003) 149–173
22. Chin, F.Y.L., Deng, X., Fang, Q., Zhu, S.: Approximate and dynamic rank aggregation. Theor. Comput. Sci. **325**(3) (2004) 409–424

23. Liu, T., Moore, A.W., Gray, A., Yang, K.: An investigation of practical approximate nearest neighbor algorithms. In Saul, L.K., Weiss, Y., Bottou, L., eds.: Advances in Neural Information Processing Systems 17. MIT Press, Cambridge, MA (2005) 825–832
24. Weinberger, K., Blitzer, J., Saul, L.: Distance metric learning for large margin nearest neighbor classification. In Weiss, Y., Schölkopf, B., Platt, J., eds.: Advances in Neural Information Processing Systems 18. MIT Press, Cambridge, MA (2006)