Study on Meta-Learning Approach Application in Rank Aggregation Algorithm Selection

Alexey Zabashta[1], Ivan Smetannikov[1], and Andrey Filchenkov[1]

[1] ITMO University, St. Petersburg, Russia
{zabashta}@rain.ifmo.ru, {smeivan, aaafil}@mail.ru

Abstract. Rank aggregation is an important task in many areas, nevertheless, none of rank aggregation algorithms is best for all cases. The main goal of this work is to develop a method, which for a given rank list finds the best rank aggregation algorithm with respect to a certain optimality criterion. Two approaches based on meta-feature description are proposed and one of them shows promising results.

Keywords: meta-learning, rank aggregation, permutations, algorithm selection.

1 Introduction

In many fields where multiple ranking algorithms are applied in practice such as computational biology, web search, or social choice, the important task of rank aggregation arises. A ranked list of objects is a permutation on these objects. Formally, the task of rank aggregation consists in finding a permutation π for a given permutation list Q, which minimizes the error function $E_\mu(\pi, Q)$, depending on a metric μ.

The problem of finding the best possible resulting rank is usually NP-hard, approximate algorithms are used, and they show different quality of results. Therefore, the problem of algorithm selection arises.

One of the possible solutions of this problem is the meta-learning approach [1]. Meta-learning systems were developed to solve different machine learning tasks, but to the best of our knowledge, the problem of rank aggregation algorithm selection has never been considered in scientific literature.

The main goal of this work is to develop an algorithm for rank aggregation algorithm selection. The proposed approach is based on meta-learning.

2 Algorithms and approaches

The *basic meta-features approach* (BMFA) for each μ-th metric looks over all possible pairs of permutations from the input permutation list Q and then constructs a sequence $X_\mu = \{\mu(a, b) | a, b \in Q\}$.

After that it mines statistic characteristics from each sequence X_μ as meta-features:

$$\text{features}(Q) = \bigcup_{\mu \in M} \begin{Bmatrix} \text{Min}(X_\mu) & \text{E}(X_\mu) & \text{Skew}(X_\mu) \\ \text{Max}(X_\mu) & \text{Var}(X_\mu) & \text{Kurt}(X_\mu) \end{Bmatrix},$$

where *M* is the set composed of the following seven metrics: described in [2, 3]: the *Manhattan distance*, the *Euclidean distance*, the *Chebyshev distance*, the *Cayley distance*, the *Kendall tau rank distance*, and the *Ulam distance*, the *Canberra distance*.

The *accelerated meta-features approach* (AMFA) aggregates the input permutation list into a single permutation c by means of the faster method — we use the Borda count. Then we construct a sequence X_μ from the distances between c and permutations from the input list $X_\mu = \{\mu(\pi, c) \mid \pi \in Q\}$. Then we mine features in the same way as in BMFA.

The *algorithm for Generating Permutation List* (AGPL) uses parameters α, β and we use three different algorithms for single permutation generation. The *Hidden Variable Approach* (HVA) describes a permutation list with α and β, with which it was generated. It can be applied only to generic data.

3 Experiments and results

In this paper we use four popular rank aggregation algorithms [2, 4]: *Borda count*, the *Copeland's Score*, the *Markov chain method,* and the "*Pick a perm"* method.

For experiments with generic data we generate permutation lists of the length 36 with 25 elements. For real-world experiments we use popular benchmark datasets for rank aggregation "LETOR4.0 MQ2007-agg".

Table 1 shows the F_1-measure for different classifiers built by the introduced approaches on three generic datasets and one real-world dataset. The table shows that the AMFA outperforms all the other approaches.

Table 1. Comparison of approaches by F_1-measure on the generic and real-work datasets.

	AGPL-A		AGPL-B		AGPL-C		Real-world	
	HVA	AMFA	HVA	AMFA	HVA	AMFA	BMFA	AMFA
IBk	0.534	**0.561**	0.400	**0.448**	0.485	**0.557**	0.367	**0.447**
J48	0.535	**0.545**	0.406	**0.413**	0.486	**0.548**	0.346	**0.413**
LogitBoost	0.529	**0.590**	0.401	**0.461**	0.487	**0.593**	0.399	**0.468**
NaiveBayes	0.522	**0.542**	**0.400**	0.393	0.476	**0.509**	**0.400**	0.396
SMO	0.386	**0.608**	0.382	**0.516**	0.431	**0.602**	0.407	**0.489**

4 Conclusion and future work

In this work we have proposed three approaches, and one of them, namely AMFA, has shown promising results on both types of on generic and real-world data. Based on this work we can conclude that meta-learning could be applied for best aggregation algorithm prediction, but current results may be improved.

In our future work we would try to: use more rank aggregation algorithm models and algorithms; apply feature selection algorithms; introduce wider generic data class, and test on other real-world data; create quality measure depending also on execution time and explore its behavior; predict best strategies for stochastic rank aggregation algorithms; create new meta-features, including task-specific meta-features.

This work was partially financially supported by the Government of Russian Federation, Grant 074-U01.

References

1. P. Brazdil, C. Giraud Carrier, C. Soares, R. Vilalta, *Metalearning. Applications to Data Mining*. Springer, 2009.
2. A. Burkovski, L. Lausser, J.M. Kraus, H.A. Kestler, *Rank Aggregation for Candidate Gene Identification*, M. Spiliopoulou et al. (eds.) Data Analysis, Machine Learning and Knowledge Discovery, Springer International Publishing, 285–293, (2014).
3. M. Deza, T. Huang, '*Metrics on Permutations, a Survey*', Journal of Combinatorics, Information and System Sciences, 23, 173–185, (1998).
4. C. Dwork, R. Kumar, M. Noar, D. Sivakumar, *Rank aggregation methods for the web*, 10th International Conf. on the World Wide Web, ACM Press and Addison Wesley, 613–622, (2001).