

# Information Retrieval using Machine learning

G Lavanya<sup>1</sup>, M Sai Kumar Reddy<sup>2</sup>, MD Thamjeed<sup>3</sup>

1,2 Department of Computer Science and Engineering, CMR Technical Campus, Medchal

Email: lavanya.g@gmail.com<sup>1</sup>

Email: saikumarchintu23@gmail.com<sup>2</sup>

Email: 167r1a05f7@cmrtc.ac.in<sup>3</sup>

**Abstract.** The Ranking is one of the big issues in various information retrieval applications (IR). Various approaches to machine learning with various ranking applications have new dimensions in the field of IR. Most work focuses on the various strategies for enhancing the efficiency of the information retrieval system as a result of how related questions and documents also provide a ranking for successful retrieval. By using a machine learning approach, learning to rank is a frequently used ranking mechanism with the purpose of organizing the documents of different types in a specific order consistent with their ranking. An attempt has been made in this paper to position some of the most widely used algorithms in the community. It provides a survey of the methods used to rank the documents collected and their assessment strategies.

## 1. Introduction

Ranking is the main issue in the area of information retrieval. To rank all the significant records from the given corpus for a given client question as per their pertinence is the focal issue in the field of Information Retrieval (IR). In machine Learning, for positioning relevance and similarity based on ranking “Learning to Rank” approach is widely used. Learning-to-rank framework uncommonly utilizes the supervised machine learning algorithms and finds the best request of a rundown as indicated by their inclinations, rank or score [1]. Most of studies in learning to rank focused on generation of new model for different types of data's and different applications such as recommendation system, Web information retrieval, pattern Matching. While creating models generate appropriate feature vector is important focusing area of in information retrieval. The use of machine learning approaches for ranking make it possible to find out relevance between the relevant documents in context of given user query and place them in order of their relevance on the top of first non-relevant document in the list. Machine learning models for ranking is categories into two types. First one is scoring function and second one type of loss function [2]. Scoring functions includes gradient boosting tree [3], neural net [4], SVM [5]. SVM and boosting trees mostly applied on multiclass classification problems. Neural nets are used in variety of Information retrieval task. Mostly preferred when data is very large. It used on document and query using distributed representation. Loss functions are used as integral part of learning to rank model. Most of ranking functions works for optimization of loss function. Various documents feature are accepted as input and generate appropriate score based on used model. This loss function approached divided into three different categories Pointwise approaches [6], pairwise approaches [4], and Listwise approaches [7]. Major challenges for leaning to rank algorithm are include first one is mismatched between correct order in training and actual rating which leads the generation of proper loss function for rating and for ordering

of predicted score. Second one is Feature selection which provides representative enough to generate the scores using model. Usually, learning-to-rank utilizes the ranking function with the help of training data and prediction on test data and generate ranking. Performance evaluation of learning-to-rank models is done by using the loss function. Loss function computes deference between prediction and ground truth [8]. To improve the performance of learning with large amount of training data, learning paradigm with semi-supervised, active learning is used.

The paper is structured as follows, in section II describe the approaches used in learning-to-rank framework which categorizes into main three approaches based on the input taken for processing and loss function. Section III provides overview of various measures used for learning to rank. Section IV elaborates different learning-to rank paradigms which used to improved the performance of learning. Section V provides review on applications of learning to rank.

## 2. Approaches in LETOR

Learning to rank has three main categories: Pointwise approaches, Pairwise approaches, and Listwise approaches based on the input and loss function. They are identifying by the loss function in the specified information retrieval task using machine learning.

### 2.1 Pointwise approach

Pointwise approaches look at a single document at a time using classification or regression or ordinal regression to discover the best ranking for individual results. The scoring function is typically trained on individual documents one at a time. They effectively accept the single document and train a classifier on it to predict how important it is for the current query. By simply sorting the result list by these document scores, the final ranking is achieved. The function to be learned  $f(q,D)$  is simplified as  $f(q,d_i)$ . That is, the relevance of each document given a query is scored independently. These are able to learn the ranking function  $f(q, d_i)$  using the provided relevance value as real-valued scores (for regression), non-ordered categories (for classification) and ordered categories (for ordinal regression).

**Table 1.** Pointwise approaches.

Name of Method	Learning Type	Methods used
OPRF[9] (Optimal Polynomial Retrieval Function)	Supervised	Polynomial Regression
SLR [10]	Supervised	Stage Logistic Regression
Pranking [11]	Supervised	Ordinary Regression
McRank [12]	Supervised	Multiple Classification
CRR [13]	Supervised	Stochastic gradient decent

It is Simplicity. Existing ML models are ready to apply approach used for ranking but it has following disadvantage [8].

- Single object at single instance is considered. It predicts relative order amongst the object.
- Loss function dominated by query in case of large dataset.
- Position of document cannot be predicted by using loss function.
- The result is usually sub-optimal due to not utilizing the full information in the entire list of matching documents for each query.
- Explicit pointwise labels are required to constitute the training dataset.

## 2.2 Pairwise approach

Pairwise approaches accept pair of document together as instance. Learning and problem formulation of learning to rank is classification task mostly to find the pair with higher ranks. In Learning we uses pointwise scoring function  $f(q, d_i)$  and training samples are constructed by pairs of documents within the same query. The Pairwise approaches compares the relation of every two documents, then it rank all the documents by comparing higher-lower pair based on the ground truth. The primary objective is to minimize the number of instances where the pair of outcomes is in the wrong order compared to the ground truth and produce the pair's labels.

Suppose given the first query  $q_1$ , with  $y_1=0$  (totally irrelevant) for  $d_1$  and  $y_2=2$  (highly relevant) for  $d_2$ , then we have a new label  $y_1 < y_2$  for the document pair  $(d_1, d_2)$ .

pointwise function  $f(q, d_i)$  uses following to score difference probabilistically.

$$\Pr(i > j) \equiv \frac{1}{1 + \exp(-(s_i - s_j))} \quad (1)$$

**Table 2.** Pairwise Approaches.

Name of Method	Learning Type	Methods used
MART [14]	Supervised	Gradient Boosting Machine, Finds strong learner by using gradient decent.
Ranking SVM [15]	Supervised	Uses clicks and rough logs
Rank Boost[16]	Supervised	Boosting
Rank Net [17]	Supervised	Neural Network with gradient Descent
IRSVM [18]	Supervised	Query level normalization in the loss function
LamdaRank [19]	Supervised	Ranknet with backpropagation neural networks
Sortnet [20]	Supervised	Adaptive Ranker ordered by neural network
Direct Ranker [21]	Supervised	Generalized version of Rank net

Pairwise approach are preferred over pointwise approach because they won't needed explicit pointwise labels Only pairwise preferences are consider but few drawback are as follow[8],

- Relative information in the feature space samples with different documents in the same query is still not fully exploited.
- Increase the training complexity for large number of dataset only.
- The imbalanced allocation of the number of documents or object for the question considered or in query.
- The noisy relevance label on the single documents.
- For multiple ordered relevance judgment, the relevance judgment results in the loss of data with finer granularity when transforming them into a relevance pair.

### 2.3 Listwise Approach

Listwise approach compare the relevance of list of documents instead of providing one rank score for a particular or a single document as in Pointwise approaches method. A listwise approach tries to decide the optimal ordering of an entire list of documents. Listwise approaches use probability models to minimize the ordering error by using permutation probability given a ranking list.

Let's Consider  $\pi$  as a permutation for a given list which may have  $n$  number of documents,  $\phi(s_i)=f(q,d_i)$  as ranking function  $s_i$  given query  $q$  and document  $i$ . The probability of having a permutation  $\pi$  can be calculated as follows

$$Pr(\pi)=\prod_{i=1}^n \frac{\phi(s_i)}{\sum_{k=i}^n \phi(s_k)} \quad (2)$$

Where  $\phi(\cdot)$ , may be any exponential function.

**Table 3.** Listwise Approaches.

Name of Method	Learning Type	Methods used
SoftRank [22]	Supervised	Gradient Boosting Machine, Finds strong learner by using gradient decent.
ListNet [23]	Supervised	Uses clicks and rough logs
AdaRank [24]	Supervised	Boosting
BoltzRank [25]	Supervised	Neural Network with gradient Descent
ListMLE [26]	Supervised	Query level normalization in the loss function
RankCosign [8]	Supervised	Ranknet with backpropagation neural networks
ESRank [27]	Supervised	Adaptive Ranker ordered by neural network
FastAP [28]	Supervised	Generalized version of Rank net
Multiberry [29]	Supervised	Gradient Boosting Machine, Finds strong learner by using gradient decent.

Listwise approaches are quite complex compared to the pointwise or pairwise approaches but better approach for ranking task. But the few disadvantage of listwise approached is[8]

- The difficulty of training for the listwise method is very high. Scoring function is mostly consider as pointwise, which could be sub-optimal.
- The optimization of measure specific loss function is not trivial.
- No guaranteed that one can really find their optima in evolutionary measure

### 3. Evaluation Measures

Several evolutionary metrics have been proposed and commonly used in the evaluation of a ranking model are as given bellow. Based on the relevance they are divided into Binary Relevance and Graded Relevance. By using evolution metric we can find how well ranking model learn and perform therefore it is formulated as optimization problem with respect to metric.

#### 3.1 Binary Relevance

##### 3.1.1 Precision @k

Ratio of relevance documents with retrieved documents. precision at  $k$  given a query  $P@k(q)$  is as

$$P@k(q) \equiv \frac{\sum_{i=1}^k ri}{k} \quad (3)$$

### 3.1.2 Mean Average Precision (MAP)

First we calculate precision at k given a query  $P@k(q)$  is as mention above. Then we calculate the average precision given a query  $AP(q)$  at k items as:

$$AP(q)@k \equiv \frac{1}{\sum_{i=1}^k ri} \sum_{i=1}^k P@k(q) \times ri \quad (4)$$

Mean Average Precision is just the mean of  $AP(q)$  for all queries:

$$MAP \equiv \frac{\sum_{q=1}^Q AP(q)}{Q} \quad (5)$$

### 3.1.3 Mean Reciprocal Rank (MRR)

This method assumes each query having reciprocal rank. RR fined the first correctly predicted relevant item in a list. Suppose reciprocal rank is  $ri$  then the inverse of the position of that document in the rank list is Mean Reciprocal Rank and calculated as.

$$MRR \equiv \frac{1}{Q} \sum_{i=1}^Q \frac{1}{ri} \quad (6)$$

## 3.2. Graded Relevance

### 3.2.1 Normalized Discounted Cumulative Gain (NDCG)

A new assessment measure called Normalized Discounted Cumulative Gain[16], which can accommodate several levels of relevant judgments, has recently been proposed. NDCG follows two rules when evaluating a ranking list:

1. Documents of high significance are more important than documents of marginal relevance.
2. The lower document ranking status (of any importance level) is the lower value for the user, since the user is less likely to be investigated.

According to the above rules, the NDCG value of a ranking list at position n is calculated as follows:

First we define Discounted Cumulative Gain at position as:

$$DCG@k \equiv \sum_{i=1}^k \frac{2^{li-1}}{\log 2(i+1)} \quad (7)$$

where  $li$  is the grading of relevance at rank  $i$ , Normalized DCG is then defined as:

$$NDCG@k = \frac{DCG@k}{IDCG@k} \quad (8)$$

where  $IDCG@k$  is the Ideal  $DCG@k$  given the result. It is the  $DCG@k$  calculated by re-sorting the given list by its true relevance labels. That is,  $IDCG@k$  is the maximum possible  $DCG@K$  value one can get given a ranking list.

### 3.2.2 Expected Reciprocal Rank

In this method the user will be satisfied up to the  $r$ th ranked document in the list and will not go further in the rank list. It can be defined as

$$ERR = \sum_{r=1}^n \frac{1}{r} P(r) \quad (9)$$

Where  $P(r)$  is the probability that user will stop a position  $r$  and will not check any document after that[21].

#### 4. Paradigm used in learning to Rank

In recent year more focus is given on parameter findings and optimization using different paradigm which include semisupervised, reinforcement learning, deep learning and parallel computing.

##### 4.1 Semi supervised Learning

It work on learning from both the type of data i. e. Small number of label data and large collection of unlabelled data [10-11]. Classification task in information retrieval uses the semi supervised approaches has three classes which includes training, feature extraction, and regularization.

**Table 4.** Semi supervised Learning Methods.

Name of Method/Algorithm	Methods used	Applications
SSLamdaRank[30]	Direct optimization NDCG and mean average precision metrics and increased information retrieval accuracy over LambdaRank.	Training done on yahoo Database, Feature Extraction is done by considering neighbourhood relations where regularizer exploits structure in the unlabeled data.
semi-supervised learning to rank	Pseudo labels are generated from selected queries for performance gain.	Training done on LETOR dataset and query features are used by query-quality predictor to uncertain data.
SSLPP [31]	Model is mostly giving the manifold dimensionality reduction algorithm with learning to rank method by using graph method.	MSRA-MM 1.0 and MSRA-MM 2.0 image datasets are used with similarity measure between features regularization is done by using graphs.
RankRLS[32]	Regulation is based on least squares method	Affect the computation cost.

##### 4.2 Reinforcement learning

**Table 5.** Reinforcement Learning Methods.

Name of Method/Algorithm	Methods used	Applications
MA-RDPG [33]	Multi-agent reinforcement learning model used where multiple agents work collaboratively to optimize the overall performance	Model evaluated on E-commerce platform which having a centralized critic, agents, and various communication component to share and encode the messages.
QRC-Rank[34]	two phase learning model calculated click-through features where agent tries to find the suitable label for a given state, with respect to a visited query-document pair.	Q-Learning and SARSA algorithms are modified click-through features for information retrieval through Web search engines.
RRLUFF [35]	Agent learning system for the selection of documents sorted as ranking done by agent.	Web documents ranking as problem solve by RRLUFF algorithm which combined $\epsilon$ -greedy and Roulette Wheel methods.
MDPRank[36]	Optimizes information retrieval measures with Monte-Carlo stochastic gradient ascent with policy gradient	Documents ranking with optimization of features using Reinforcement

---

algorithm of REINFORCE was used to train the model parameters.

---

#### 4.3 Deep Learning

**Table 6.** Reinforcement Learning Methods.

Name of Method/Algorithm	Methods used	Applications
RankTxNet[37]	Uses deep network of self-attention based transformer and bidirectional sentence encoder on Sentence Ordering and Order Discrimination.	Extension of BERT algorithm with the combination of feed forward network for decoding optimized the sentence ordering applications
DeepQRank[38]	Customizing the reward function and neural network of deep q-learning. Also developed polyak averaging-like method	Extension of MDPRank algorithm with Policy gradient and deep q-learning method to MDP representation problem.
DCN-V2[39]	Propose a new model DCNV2 to learn explicit and implicit feature of query with low-rank techniques to approximate feature crosses to improve performance.	Mostly applied on web-scale learning to rank systems with optimization of rank matrices.
Sortnet[40]	Neural network was used as comparator to elect the most informative patterns in the training set.	Different Preference learning application was solved as Ranking of objects according to users and need.

#### 4.4 Parallel Computing

**Table 7.** Reinforcement Learning Methods.

Name of Method/Algorithm	Methods used	Applications
PROFL[41]	Explore the parallel algorithms and for the GPU implementations is done by selective sampling (PRSS) in on-demand learning.	Decreased the training time using selective sampling for customized ranking. Presented use of parallel algorithms and implementations of GPU for speedups up.
PLtR-B and PLtR-N[42]	Used a parallel SGD scheme which is lock-free to improve the efficiency.	Used mostly in collaborative filtering with learning from streaming user feedback efficiently. Both methods combined with adaptive gradient update methods to increase the learning rate.



## 5. Applications

Web Information retrieval is popular application with the use of the learning-to-rank. Machine learning with information systems reduces the drawback of conventional ranking systems which increased engagement of learning to rank algorithm in many application. Few applications are listed below.

- Recommender system: With the development of various recommendations and increasing E-commerce websites required personalized recommendations with preferences of the use. It is primary need to provide the recommendation. Learning-to-rank provides conventional rating prediction for recommendation.
- Stock portfolio selection: Prediction is used in most of the application. Learning to rank algorithm provides the robustness to the system. All the systems uses past historical data and with the help of machine learning and ranking system loss error can be reduce. Qiang Song[43] and et. al. Developed an stock portfolio selection by combining the features of ListNet and RankNet algorithm with more reliable predictions.
- Message auto reply: Auto reply is end to end method for generating specific type of content messages which having response selection, response set generator, diversity and triggering models components. Mostly used predictive response or target response. Learning to rank methods are used for prediction responses with low error rate.
- Image to text: Image ranking employed an image content description such that similar images can be retrieved. Fabio et. al[44] proposed learning to rank algorithms with three different techniques which improve the ranking of documents.

## 6. Conclusion

The use of machine learning methods for IR ranking is becoming an evolving issue in the research based on ranking. This paper summarises the various learning methods used to learn to rank models. In learning to rank problem, we look at all the different learning methods along with their some of the most widely used algorithms and evaluation steps. While some of the algorithms in search engines have been implemented, all the queries can still not be answered by an algorithm. There are three main groups of the supervised learning system for ranking. The first is the pointwise method, which reduces the rating on each single document to regression, classification, or ordinal regression.. Second is the pair method, which essentially formulates ranking on each document pair as a classification problem. The third is the listwise method, which considers ranking as a new problem and attempts to optimise a measure-specific or non-measure-specific loss function, which is described on all query-related documents. It can be inferred from the above all equations and learning that the listwise approaches are to have better output and that particular issues can be easily addressed in question.

## 7. Acknowledgment

We thank CMR Technical Campus for supporting this paper titled with "Information Retrieval using Machine learning", which provided good facilities and support to accomplish our work. Sincerely thank to our Chairman, Director, Deans, Guide and Faculty Members for giving valuable suggestions and guidance in every aspect of our work.



## 8. References

- [1] H. Li, "A short introduction to learning to rank," *IEICE Trans. Inf. Syst.*, vol. 94, no. 10, pp. 1854-1862, 2011.
- [2] Ruixin Wang, Kuan Fang, Rikang Zhou, Zan Shen and Liwen Fan, "SERank: Optimize Sequencewise Learning to Rank Using Squeeze-and-Excitation Network", *Journal in arXiv preprint*, 2020.
- [3] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu "Lightgbm: A highly efficient gradient boosting decision tree". In *Advances in Neural Information Processing Systems*. 3146–3154, 2017.
- [4] Christopher JC Burges. "From Ranknet to Lambdarank to Lambdamart: An overview", *Learning* 11, 23-581, 2010.
- [5] Thorsten Joachims. "Training linear SVMs in linear time", In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data*, 2006.
- [6] Wei Chen, Tie-Yan Liu, Yanyan Lan, Zhi-Ming Ma, and Hang Li., "Ranking measures and loss functions in learning to rank". In *Advances in Neural Information Processing Systems*, 2009, 315–323.
- [7] Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. "Listwise approach to learning to rank: theory and algorithm". In *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, 1192–1199.
- [8] Ashwini Rahangdale et. al. "Machine Learning Methods for Ranking", *International Journal of Software Engineering and Knowledge Engineering* Vol. 29, No. 6, PP 729–761, 2019.
- [9] N. Fuhr. "Optimum polynomial retrieval functions", In *Proceedings of the 12th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '89)*, 1989.
- [10] Cooper, W. et al. "Probabilistic Retrieval in the TIPSTER Collections: An Application of Staged Logistic Regression." *TREC* (1992).
- [11] Koby Crammer and Yoram Singer, "Pranking with ranking", In *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic (NIPS'01)*. MIT Press, Cambridge, MA, USA, 641–647, 2001.
- [12] Li, Ping and Burges, Chris J.C. and Wu, Qiang, "Learning to Rank Using Classification and Gradient Boosting", *Advances in Neural Information Processing Systems*, 2008.
- [13] D. Sculley., "Combined regression and ranking", In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '10)*. Association for Computing Machinery, New York, NY, USA, 979–988, 2010.
- [14] Monteiro, Antonio, Jorge, Ferreira, da, Silva. "Multiple additive regression trees: a methodology for predictive data mining for fraud detection", *Calhoun*, 2002.
- [15] Joachims, T., "Optimizing Search Engines using Clickthrough Data" (PDF), *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining*, 2002.
- [16] Yoav Freund, Raj Iyer, Robert E. Schapire, Robert E. Schapire "An Efficient Boosting Algorithm for Combining Preference", *Journal of Machine Learning Research*, 933-969, 2003.
- [17] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. "Learning to rank using gradient descent". In *Proceedings of the 22nd international conference on Machine learning (ICML '05)*. Association for Computing Machinery, New York, NY, USA, 89–96, 2005.
- [18] Yunbo Cao, Jun Xu, Tie-Yan Liu, Hang Li, Yalou Huang, and Hsiao-Wuen Hon, "Adapting Ranking SVM to Document Retrieval", *SIGIR* 2006.
- [19] Burges, Christopher & Ragno, Robert & Le, Quoc., "Learning to Rank with Non smooth Cost Functions", *Advances in Neural Information Processing Systems* 19, 193-200, 2006.
- [20] L. Rigutini, T. Papini, M. Maggini, and F. Scarselli, "SortNet: Learning to Rank by a Neural Preference Function", *IEEE Transactions on Neural Networks*, 1368–1380, 2011.
- [21] Köppl M., Segner A., Wagener M., Pensel L., Karwath A., Kramer S. "Pairwise Learning to Rank by Neural Networks Revisited: Reconstruction, Theoretical Analysis and Practical Performance." In *Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2019. Lecture Notes in Computer Science*, vol 11908. Springer, Cham, 2020.
- [22] Michael Taylor, John Guiver, Stephen Robertson, and Tom Minka., "SoftRank: optimizing non-smooth rank metrics". In *Proceedings of the 2008 International Conference on Web Search and Data Mining (WSDM '08)*. Association for Computing Machinery, New York, NY, USA, 77–86, 2008.
- [23] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li, "Learning to rank: from pairwise approach to listwise approach", In *Proceedings of the 24th international conference on Machine learning (ICML '07)*. Association for Computing Machinery, New York, NY, USA, 129–136, 2007.
- [24] Jun Xu and Hang Li., "AdaRank: a boosting algorithm for information retrieval", In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '07)*. Association for Computing Machinery, New York, NY, USA, 391–398, 2007.
- [25] Maksims N. Volkovs and Richard S. Zemel. "BoltzRank: learning to maximize expected ranking gain". *Proceedings of the 26th Annual International Conference on Machine Learning*. Association for Computing Machinery, New York, NY, USA, 1089–1096, 2009.
- [26] Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li, "Listwise approach to learning to rank: theory and algorithm", In *Proceedings of the 25th international conference on Machine learning (ICML '08)*. Association for Computing Machinery, New York, NY, USA, 1192–1199, 2008.

- [27] Osman Ali Sadek Ibrahim and Dario Landa-Silva, "ES-Rank: evolution strategy learning to rank approach" In Proceedings of the Symposium on Applied Computing (SAC '17). Association for Computing Machinery, New York, NY, USA, 944–950, 2017.
- [28] F. Cakir, K. He, X. Xia, B. Kulis and S. Sclaroff, "Deep Metric Learning to Rank," IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, pp. 1861-1870, 2019.
- [29] Stanton, Andrew & Ananthram, Akhila & Su, Congzhe & Hong, Liangjie, "Revenue, Relevance, Arbitrage and More: Joint Optimization Framework for Search Experiences in Two-Sided Marketplaces", 2019
- [30] Chapelle, O., Wu, M. "Gradient descent optimization of smoothed information retrieval metrics". Inf Retrieval 13, 216–235, 2010.
- [31] Zhong Ji, Yanwei Pang, Yuqing He, and Huanfen Zhang, "Semi-supervised LPP algorithms for learning-to-rank-based visual search reranking". Inf. Sci. 302, C (May 2015), 83–93, 2015.
- [32] Airola, Antti & Pahikkala, Tapio & Salakoski, Tapio., "Large Scale Training Methods for Linear RankRLS", Proceedings of the ECML/PKDD, 2010.
- [33] Jun Feng, Heng Li, Minlie Huang, Shichen Liu, Wenwu Ou, Zhirong Wang, and Xiaoyan Zhu, "Learning to Collaborate: Multi-Scenario Ranking via Multi-Agent Reinforcement Learning". In Proceedings of the 2018 World Wide Web Conference (WWW '18). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 1939–1948, 2018.
- [34] Keyhanipour, A. H. et al., "Learning to rank with click-through features in a reinforcement learning framework", International Journal of Web Information Systems, vol. 12, no. 4, pp. 448–476, 2016.
- [35] Derhami, V., Paksima, J., & Khajeh, H. "RRLUFF: Ranking function based on Reinforcement Learning using User Feedback and Web Document Features", Journal of AI and Data Mining, 7, 421–442, 2019.
- [36] Zeng Wei, Jun Xu, Yanyan Lan, Jiafeng Guo, and Xueqi Cheng, "Reinforcement Learning to Rank with Markov Decision Process". In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17). Association for Computing Machinery, New York, NY, USA, 945–948, 2017
- [37] Kumar, P., Brahma, D., Karnick, H., & Rai, P. "Deep Attentive Ranking Networks for Learning to Order Sentences". Proceedings of the AAAI Conference on Artificial Intelligence, 34(05), 8115–8122, 2020.
- [38] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Jingfang Xu, and Xueqi Cheng, "DeepRank: A New Deep Architecture for Relevance Ranking in Information Retrieval", In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management (CIKM '17), Association for Computing Machinery, New York, NY, USA, 257–266, 2017.
- [39] Ruoxi Wang, Rakesh Shivanna, Derek Z. Cheng, Sagar Jain, Dong Lin, Lichan Hong, Ed H. Chi, "DCN V2: Improved Deep & Cross Network and Practical Lessons for Web-scale Learning to Rank Systems", 2020.
- [40] L. Rigutini, T. Papini, M. Maggini and F. Scarselli, "SortNet: Learning to Rank by a Neural Preference Function," in IEEE Transactions on Neural Networks, vol. 22, no. 9, pp. 1368–1380, Sept. 2011.
- [41] e Freitas, MF, Sousa, DX, Martins, WS, Rosa, TC, Silva, RM, Gonçalves, MA. "Parallel rule-based selective sampling and on-demand learning to rank". Concurrency Computat Pract Exper. 2019; 31e4464
- [42] Murat Yagci, Tevfik Aytekin, and Fikret Gorgen, "On Parallelizing SGD for Pairwise Learning to Rank in Collaborative Filtering Recommender Systems.", In Proceedings of the Eleventh ACM Conference on Recommender Systems (RecSys '17). Association for Computing Machinery, New York, NY, USA, 37–41, 2017.
- [43] Qiang Song, Anqi Liu, Steve Y. Yang, "Stock portfolio selection using learning-to-rank algorithms with news sentiment", Neurocomputing, Volume 264, Pages 20–28, 2017.
- [44] Hu, H.Y., Zheng, W.F., Zhang, X., Zhang, X., Liu, J., Hu, W.L., Duan, H.L., Si, J.M. , "Content-based gastric image retrieval using convolutional neural networks", International Journal of Imaging Systems and Technology, 2020.