International Journal of Technology Research and Management
ISSN (Online): 2348-9006
Vol 4 Issue 9 September 2017

Ranking and Feedback Model in Search Engines

M. Supriya
Associate Professor, Dept of Information Technology, Swami Vivekananda Institute of Science and Technology, Secundrabad, Telangana, 500003, India
supriyasamuel@yahoo.com

Abstract: *With the unstable emergence of vertical search domains, applying the broad-based ranking model in a straight line to different domains is not any more desirable owed to domain differences, whereas building a unique ranking model for each domain is mutually laborious for labeling data and an extremely time consuming process for training models. In this article, we can begin these difficulties by proposing a regularization-based algorithm called ranking adaptation-SVM (RA-SVM), from the beginning to end we can also get used to an existing ranking model to a new domain, so that the amount of labeled data and the training cost is minimized, even though the performance is still certain. Our algorithm only requires the forecast from the existing ranking models, rather than their internal representations or the data from auxiliary domains. In addition, we take for granted that documents having similar in the domain-specific feature space should have consistent rankings, and add some constraints to control the margin and slack variables of RA-SVM adaptively. In conclusion, ranking adaptability measurement is proposed to quantitatively estimate if an existing ranking model can be adapted to a new domain. Experiment performed over LETOR and two large scale data sets crawled from a commercial search engine demonstrate the applicability's of the proposed ranking adaptation algorithms and the ranking adaptability measurement.*

Keywords: *Ranking, Support Vector Machines, Domain adoption, Learning to rank.*

1. INTRODUCTION

Learning to rank is a variety of learning base information retrieval method, definite in learning a ranking model with a few documents labeled with their relevancies to some queries, where the model is confidently able of ranking the documents returned to an arbitrary new query by design. Depending on the range of machine learning methods, e.g., Ranking type, with subscripts and superscripts in a slightly smaller font size. This is acceptable. SVM the learning to rank algorithms has already shown their capable performances in the information recovery, particularly Web search. However, as the materialization of domain-specific search engines, more special treatment have moved from the broad based search to definite verticals, for hunt information limitation to a certain region. dissimilar vertical search engines deals with different themes, types of document or domain-specific features. For example, nearly all of the search engine should obviously be specialized in terms of its contemporary focus, whereas a music, image or video search engine would concern only the credentials in a particular formats, since currently the broad-based and vertical search engines are habitually based on text search techniques, the ranking model educated for broad- based can be utilized directly to rank the documents for the verticals. For example, the majority of current image search engines only make use of the text information accompanying images as the ranking features, such as the Term frequency (TF) of query word in image title, surrounding text, alternative text ,anchor text, URL and so on. As a result, Web images are in fact treats as the text-based documents that distribute related ranking features as the document or Web page ranking, and text based ranking model can be applied here directly. On the other hand, the broad-based ranking model is built upon the data from multiple domains, and therefore cannot simplify well for a particular field with special search intentions. In addition, the broad-based ranking model can only make use of the vertical domain's ranking features that are same to the broad based domains for ranking, while the domain-specific features, such as the content features of images, videos or

music cannot be utilize directly. Those features are generally important for the semantic representation of the documents and should be make use of to build a more strong ranking model for the particular vertical. Ranking Support Vector Machines (Ranking SVM), is one of the most successful learning to rank algorithms, and is employed as the foundation of our proposed algorithm, the proposed RS_SVM does not call for the labeled training samples from the auxiliary domain, but only is ranking model fa . Such a model is more useful than the data based adaptation, for the reason that the training data from auxiliary domain may be omitted or unavailable for the copyright protection or privacy issue, but the ranking model is rather easier to obtain and access.

2. LITERATURE SURVEY

1) Manifold Regularization: A Geometric Framework for Learning from Labeled and Unlabeled Examples
AUTHORS: M. Belkin, P. Niyogi, and V. Sindhwani

We propose a family of learning algorithms based on a new form of regularization that allows us to exploit the geometry of the marginal distribution. We focus on a semi-supervised framework that incorporates labeled and unlabeled data in a general-purpose learner. Some transductive graph learning algorithms and standard methods including support vector machines and regularized least squares can be obtained as special cases. We use properties of reproducing kernel Hilbert spaces to prove new Representer theorems that provide theoretical basis for the algorithms. As a result (in contrast to purely graph-based approaches) we obtain a natural out-of-sample extension to novel examples and so are able to handle both transductive and truly semi-supervised settings. We present experimental evidence suggesting that our semi-supervised algorithms are able to use unlabeled data effectively. Finally we have a brief discussion of unsupervised and fully supervised learning within our general framework.

2) Domain Adaptation with Structural Correspondence Learning
AUTHORS: J. Blitzer, R. Mcdonald, and F. Pereira

Discriminative learning methods are widely used in natural language processing. These methods work best when their training and test data are drawn from the same distribution. For many NLP tasks, however, we are confronted with new domains in which labeled data is scarce or non-existent. In such cases, we seek to adapt existing models from a resource-rich source domain to a resource-poor target domain. We introduce structural correspondence learning to automatically induce correspondences among features from different domains. We test our technique on part of speech tagging and show performance gains for varying amounts of source and target training data, as well as improvements in target domain parsing accuracy using our improved tagger.

3) Learning to Rank Using Gradient Descent
AUTHORS: C.J.C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender

We investigate using gradient descent methods for learning ranking functions; we propose a simple probabilistic cost function, and we introduce RankNet, an implementation of these ideas using a neural network to model the underlying ranking function. We present test results on toy data and on data from a commercial internet search engine.

4) Real Time Google and Live Image Search Re-Ranking
AUTHORS: J. Cui, F. Wen, and X. Tang

Nowadays, web-scale image search engines (e.g. Google, Live Image Search) rely almost purely on surrounding text features. This leads to ambiguous and noisy results. We propose to use adaptive visual similarity to re-rank the text-based search results. A query image is first categorized into one of several predefined intention categories, and a specific similarity measure is used inside each category to combine image features for re-ranking based on the query image. Extensive experiments demonstrate that using this algorithm to filter output of Google and Live Image Search is a practical and effective way to dramatically improve the user experience. A real-time image search engine is developed for on-line image search with re-ranking: http://mmlab.ie.cuhk.edu.hk/intentsearch.

5) Boosting for Transfer Learning
AUTHORS: W. Dai, Q. Yang, G.-R. Xue, and Y. Yu

Traditional machine learning makes a basic assumption: the training and test data should be under the same distribution. However, in many cases, this identical-distribution assumption does not hold. The assumption might be violated when a task from one new domain comes, while there are only labeled data from a similar old domain. Labeling the new data can be costly and it would also be a waste to throw away all the old data. In this paper, we present a novel transfer learning framework called TrAdaBoost, which extends boosting-based learning algorithms (Freund & Schapire, 1997). TrAdaBoost allows users to utilize a small amount of newly labeled data to leverage the old data to construct a high-quality classification model for the new data. We show that this method can allow us to learn an accurate model using only a tiny amount of new data and a large amount of old data, even when the new data are not sufficient

to train a model alone. We show that TrAdaBoost allows knowledge to be effectively transferred from the old data to the new. The effectiveness of our algorithm is analyzed theoretically and empirically to show that our iterative algorithm can converge well to an accurate model.

3. EXISTING SYSTEM

The on hand broad-based ranking model gives a lot of frequent information in ranking documents, among them only a few working out samples are desirable to be label in the latest domain. From the probabilistic perception, the broad-based ranking model provides a prior knowledge, so that only a small number of labeled samples are enough for the target domain ranking model to achieve the same confidence. Hence to decrease the cost for latest verticals, how to adopt the secondary ranking models to the new target domain and to make complete use of their domain-specific features, turns into a key problem for building successful domain- specific ranking models.

4. PROPOSED SYSTEM

The planned System spotlight whether to adapt ranking models learned for the existing broad-based search or for a few verticals, to a new domain, so that the quantity of labeled data in the objective domain is reduced, yet the presentation requisite is definite, how to become accustomed to the ranking model effectively and efficiently, and how to make use of domain-specific features for more boost up the model adaptation.

The first problem is solved by means of the planned ranking adaptability measure, that quantitatively estimates whether an existing ranking model can be adapted to the new domain, and predicts the potential performance for the adaptation.

We take in hand the second problem from the regularization framework and a ranking adaptation SVM algorithm which is proposed. This algorithm is a Black box ranking model adaptation, which desires only the predictions from the obtainable ranking model, relatively than the internal representation of the model itself or the data from the auxiliary domains. With the help of black-box adaptation property, we have achieved not only the flexibility but also the efficiency.

To resolve the third problem, we suppose that documents similar in their domain specific feature space should have reliable rankings.

I. BENEFITS OF PROPOSED SYSTEM

1) Model adaptation.
2) Reducing the labeling cost
3) Reducing the computational cost

5. MODULES

DIFFERENT TYPES OF MODULES ARE:

a. Ranking model Adaptation Module.
b. Examine Ranking adaptability Module.
c. Ranking adaptation with domain specific search Module
d. Ranking Support Vector Machine Module

a. Ranking model adaptation Module

Ranking adaptation is directly related to classifier adaptation, which has shown its effectiveness for many learning problems. Ranking adaptation is comparatively additional challenging. Different classifier adaptation, primarily deals by means of the binary targets, ranking adaptation wishes to adapt the model which helps to predict the rankings for congregation of domains. In ranking the significance levels between different domains are sometimes different and necessitate to be aligned. We can get used to ranking models learned for the existing broad-based search or a number of verticals, to a new domain, so that the quantity of labeled data in the target domain is reduced while the performance requirement is still guaranteed and how to adapt the ranking model effectively and efficiently .Then how to utilize domain-specific features for further boost the model adaptation. The expected ranking adaptability deals the association stuck between the ranking lists sorted by auxiliary model prediction and the ground truth, that gives us a hint of Whether the auxiliary ranking model is adapted to the target domain, and what type of assistance it can offer. Depending on the ranking adaptability, we can execute routine model selection for determining which auxiliary models will be adapted.

b. Examine Ranking adaptability Module

Ranking adaptability extent by investigating the mutual relationship between the 2 ranking lists of a labeled queries in the target domain, i.e., the one forecast by fa and the ground-truth one labeled by human judges. obviously, if the 2 ranking lists have high optimistic correlation ,then the auxiliary ranking model fa overlap with the division of

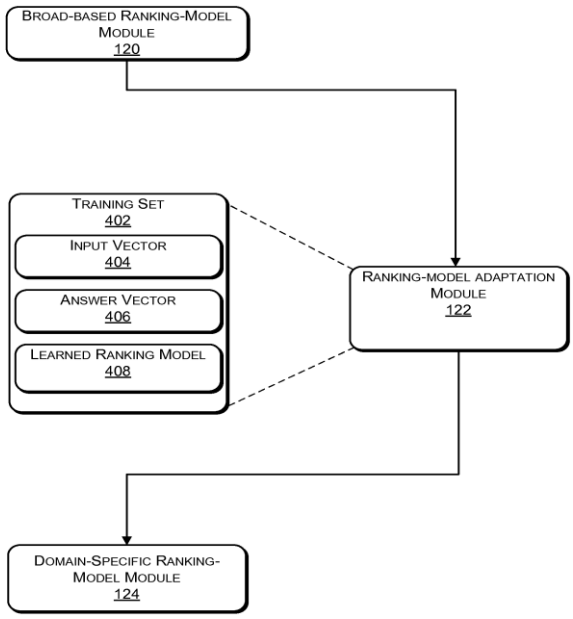

Fig1: Ranking Model Adaptation module

the corresponding labeled data, hence we can accept as true that it possesses high ranking adaptability towards the target domain, and vice versa. This is because the labeled queries are in fact randomly sampled from the target domain for the model adaptation, and can reflect the distribution of the data in the target domain.

c. Ranking adaptation with domain specific search Module

Information from various domains are also characterized by a number of domain-specific features, e.g., if we see eye to eye the ranking model learned from the Web page search domain to the image search domain, the image substance can present supplementary information to facilitate the text based ranking model adaptation. In this segment, we chat about how to operate these domain-specific features, which are usually difficult to translate to textual representations directly, to further boost the performance of the proposed RA-SVM. The essential idea of this method is to assume that documents with parallel domain-specific features ought to be assigned with similar ranking predictions. We can name the above assumption as the consistency assumption, which implies that a robust textual ranking function should perform relevance prediction that is consistent to the domain-specific features.

A Domain-specific search [engine | process] is a search [engine | process] that specifies one or more of the following five dimensions:

1. Subject areas e.g. chemical, biomedical, healthcare
2. Modality e.g. text, images, videos, sounds
3. Users e.g. a patent examiner, a professor of medicine, a project manager
4. 4. Tasks e.g. prior art patent search, technology survey, literature search, diagnosis search.
5. Tools, techniques and algorithms required to complete the tasks, e.g. query achievement restricted to specific vocabularies, cross-lingual search, opportunity to store search results.

The first two dimensions cover the information sources. Dimensions 3 and 4 cover the end users and the search tasks that they carry out, and are strongly related to work on information behavior. Dimension 5 is related to the technical aspects of the design of the search engine and supporting software. It is a 5D space, with the caveat that the dimensions are not quite orthogonal to each other. For instance, if we take the chemical domain with: images as a modality, researcher as a user and (sub-) compound search as the task.

d. Ranking Support Vector Machines Module

Ranking Support Vector Machines (Ranking SVM), is one of the most efficient learning to rank algorithms, and is here engaged as the basis of our proposed algorithm? the planned RA-SVM does not need the labeled training samples as of the auxiliary domain, but only its ranking model. Such a technique is more advantageous than data based adaptation, since the training data from the auxiliary domain may be lost or out of stock, for the official document protection or privacy issue, although the ranking model is comparatively easier to accomplish and access.

1) **Support vector Machines (Kernels)**

The SVM algorithm is implemented in practice using a kernel .The learning of the hyper plane in linear SVM is ready by transforming the trouble by using some linear algebra, which is the scope of beginning to SVM.

A powerful insight is that the linear SVM can be rephrased using the inner product of any two given observations, rather than the observations themselves. The inner product between two vectors is the sum of multiplication of each pair of input values.

For example, the internal product of the vectors [2, 3] and [5, 6] is 2*5 + 3*6 or 28.

The equation for building a forecast for a new input using the dot product between the input (x) and each support vector (xi) is calculated as follows:

f(x) = B0 + sum(ai * (x,xi))

This is an equation that calculate the inner products of a new input vector (x) with all support vectors in training the data. The coefficients B0 and ai (for each input) must be estimated from the training data by the learning algorithm.

Linear Kernel SVM

The dot-product is known as the kernel and it can be able to be rewrite as:

K(x, xi) = sum(x * xi)

The kernel defines the resemblance or the distance which measures connecting new data and the support vectors. The dot product is the similarity measure used for linear SVM or a linear kernel since the distance is a linear combination of the inputs.

Additional kernels can be used, that convert the input space into higher dimensions such as a Polynomial Kernel and a Radial Kernel. This is called the Kernel Trick.

It is advantageous to use extra complex kernels, as it allows lines to break up the classes that are curved or even more complex. This in turn be able to show the way to more precise classifiers.

2) Polynomial Kernel SVM

Instead of the dot-product, polynomial kernel can be used, for example:

K(x ,xi) = 1 + sum(x * xi)^d

Where the degree of the polynomial ought to be specified by hand to the learning algorithm. When d=1 this is same as the linear kernel. Polynomial kernel allows for curved lines in the input space.

3) Radial Kernel SVM

As a final point, we can also have a more complex radial kernel. For example:

K(x, xi) = exp(-gamma * sum((x – xi^2))

Where gamma is a parameter that have to be specified to the learning algorithm. A good default value for gamma is 0.1,where gamma is often 0 < gamma < 1. The radial kernel is enormously local and can create difficult regions within the feature space, like closed polygons in two-dimensional space.

4) Data Preparation for SVM

This section lists out some suggestions for how to best prepare your training data when learning an SVM model.

5) Numerical Inputs: SVM assumes that your inputs are numeric. If you have a definite inputs you may need to covert them to binary dummy variables (one variable for each category).

6) Binary Classification: Basic SVM as described in this post is intended for binary (two-class).

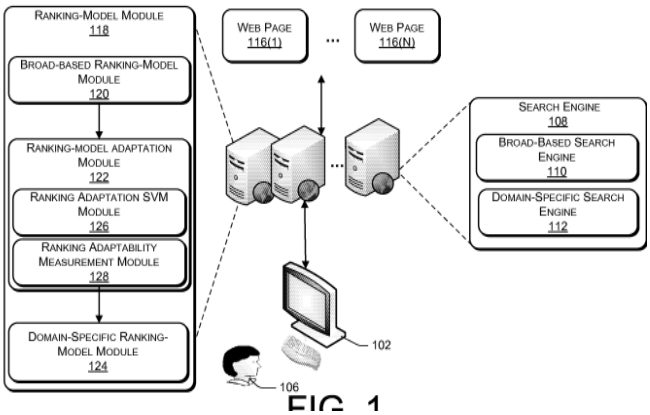

Fig 2: Block diagram: Ranking SVM

6. RELATED WORKS

There are a few other related works that are directly related to the concepts of ranking models. The one use as the beginning of the paper is Ranking SVM, that takes the structure of RA-SVM. Some of the boosting ranking models for ranking a page is Rank Boost [10].

7. FUTURE IMPROVEMENTS

Every application has its own qualities of merits and demerits. This paper has covered almost all the necessities. Additional requirements and improvements can be simply prepared since the coding is mainly planned or modular in nature. Altering the existing modules or adding new modules can append improvements. Further development can be implemented in this project . like further extended to various domains ,Image search , document retrieval ,map search can also be implemented in this.

8. CONCLUSION

As a mixture of vertical search engines emerge and the amount of verticals increases radically, a global ranking model, which is skilled over a dataset sourced from multiple domains, cannot give a sound performance for each specific domain with special topicalities, document formats and domain-specific features. Building one model for each

vertical domain is both laborious for labeling the data and time-consuming for learning the model. In this paper, we propose the ranking model adaptation, to adapt the well learned models from the broad-based search or any other auxiliary domains to a new target domain. By model adaptation, only a small number of samples need to be labeled, and the computational cost for the training process is greatly reduced. Based on the regularization framework, the Ranking Adaptation SVM (RA-SVM) algorithm is proposed, which performs adaptation in a black-box way, i.e., only the relevance predication of the auxiliary ranking models is needed for the adaptation. Based on RASVM, two variations called RA-SVM margin rescaling (RA-SVM-MR) and RA-SVM slack rescaling (RA-SVMSR) are proposed to utilize the domain specific features to further facilitate the adaptation, by assuming that similar documents should have consistent rankings, and constraining the margin and loss of RA-SVM adaptively according to their similarities in the domain-specific feature space. Furthermore, we propose *ranking adaptability*, to quantitatively measure whether an auxiliary model can be adapted to a specific target domain and how much assistance it can provide. We performed several experiments over Letor benchmark datasets and two large scale datasets obtained from a commercial internet search engine, and adapted the ranking models learned from TD2003 to TD2004 dataset, as well as from Web page search to image search domain. Based on the results, we can derive the following conclusions:

REFERENCES

[1] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. Journal of Machine Learning Research, 7(Nov):2399–2434, 2006.
[2] J.Blitzer, R. Mcdonald, and F. Pereira. Domain adaptation with structural correspondence learning. In EMNLP ’06: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Pro-cessing, pages 120–128. Association for Computational Linguistics, July 2006.
[3] C. J. C. Burges, R. Ragno, and Q. V. Le. Learning to rank with nonsmooth cost functions. In NIPS ’06: Advances in Neural Information Processing Systems, pages 193–200. MIT Press, Cambridge, MA, 2006.
[4] C. J. C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In ICML ’05: Proceedings of the 22th International Conference on Machine Learning, 2005.
[5] Z. Cao and T. yan Liu. Learning to rank: From pairwise approach to list wise approach. In ICML ’07: Proceedings of the 24th International Conference on Machine Learning, pages 129–136, 2007.
[6] J. Cui, F. Wen, and X. Tang. Real time google and live image search re-ranking. In ACM Multimedia, pages 729–732, 2008.
[7] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu. Boosting for transfer learning. In ICML ’07: Proceedings of the 24th international conference on Machine learning, pages 193–200, 2007.
[8] H. Daume, III and D. Marcu. Domain adaptation for statistical classifiers. Journal of Artficial Intelligence Research, 26:101–126, 2006.
[9] Y. Freund, R. Iyer, R. E. Schapire, Y. Singer, and G. Dietterich. An efficient boosting algorithm for combining preferences. Journal of Machine Learning Research, 4:933–969, 2003.
[10] B. Geng, L. Yang, C. Xu, and X.-S. Hua. Ranking model adaptation for domain-specific search. In CIKM ’09: Proceeding of the 18th ACM conference on Information and knowledge management, pages 197–206, 2009.

Sites Referred:

http://java.sun.com
http://www.roseindia.com/
http://www.java2s.com/