A Unified Posterior Regularized Topic Model with Maximum Margin for Learning-to-Rank[*]

Shoaib Jameel[1], Wai Lam[2], Steven Schockaert[1], Lidong Bing[3]
[1]School of Computer Science and Informatics, Cardiff University.
[2]Dept. of Systems Engineering and Engineering Management, The Chinese University of Hong Kong.
[3]Machine Learning Department, Carnegie Mellon University.
{jameels1,schockaerts1}@cardiff.ac.uk wlam@se.cuhk.edu.hk lbing@cs.cmu.edu

ABSTRACT

While most methods for learning-to-rank documents only consider relevance scores as features, better results can often be obtained by taking into account the latent topic structure of the document collection. Existing approaches that consider latent topics follow a two-stage approach, in which topics are discovered in an unsupervised way, as usual, and then used as features for the learning-to-rank task. In contrast, we propose a learning-to-rank framework which integrates the supervised learning of a maximum margin classifier with the discovery of a suitable probabilistic topic model. In this way, the labelled data that is available for the learning-to-rank task can be exploited to identify the most appropriate topics. To this end, we use a unified constrained optimization framework, which can dynamically compute the latent topic similarity score between the query and the document. Our experimental results show a consistent improvement over the state-of-the-art learning-to-rank models.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Clustering, Retrieval models, Search process*

Keywords

Learning-to-rank; Topic models; Maximum margin learning

1. INTRODUCTION

The learning-to-rank (LTR) paradigm for information retrieval (IR) consists in the use of machine learning techniques for constructing suitable document ranking functions. Documents in this context are represented as vectors of features, capturing relevance w.r.t. the given query as well as query-independent statistics such as PageRank. LTR approaches can be distinguished in how they approach the ranking problem. In particular, pointwise (e.g. [22]), pairwise (e.g. [7]), and listwise (e.g. [8, 19, 31]) approaches are commonly considered, which respectively interpret ranking as a regression problem, a binary classification problem and an optimization problem.

In this paper, we will extend a maximum margin based LTR model, which is known to perform well on this task [22, 14, 1, 7], with information derived from a latent topic model, which has already proven beneficial in many IR tasks [30, 32, 10]. Several authors have already considered the use of latent topics for LTR. A common approach to do this is to adopt a two-stage "downstream" method [34], in which an existing topic model such as Latent Dirichlet Allocation (`LDA`) [6] is used to find the latent topics in the documents and queries. One can then compute a topic-based similarity between the query and a document, e.g. using cosine similarity or a likelihood based score [30]. By adding the resulting similarity scores as an additional feature, existing LTR models can be used to learn a ranking function for these topic-enriched feature vectors. For example, such a two-stage method has been used in [27] as a comparative method. However, since the mechanisms behind discovering the topics and learning the ranking are completely decoupled, this approach is inherently sub-optimal. Indeed, similar approaches have already been shown to perform unsatisfactorily in other prediction tasks [34, 27], as errors from one stage are propagated to the next.

The mechanism we propose in this paper is significantly different from these existing two-stage "downstream" approaches, as it is based on a tight coupling of latent topic detection and maximum margin classification. Specifically, we propose a unified constrained optimization framework, which is used to find a regularized posterior distribution of the predictive function in a space defined by the pairwise maximum margin constraints. The topic model component of the composite objective function is used to find the latent dimensions of the dataset, whereas the maximum margin component is used for label prediction. The advantage of this approach is that latent topic information can be chosen so as to aid the classification of data points around the de-

[*]The work described in this paper is substantially supported by grants from the Research Grant Council of the Hong Kong Special Administrative Region, China (Project Codes: 413510 and 14203414) and the Direct Grant of the Faculty of Engineering, CUHK (Project Code: 4055034). This work is also affiliated with the CUHK MoE-Microsoft Key Laboratory of Human-centric Computing and Interface Technologies. Steven Schockaert has been supported by ERC Starting Grant 637277. This work was done when the first author was a Postdoctoral Fellow at The Chinese University of Hong Kong.

CIKM'15, October 19–23, 2015, Melbourne, Australia.
© 2015 ACM. ISBN 978-1-4503-3794-6/15/10 ...$15.00.
DOI: http://dx.doi.org/10.1145/2806416.2806482.

cision boundary of a standard pairwise classifier, and more generally to prevent misclassifications. In particular, the proposed framework is capable of obtaining additional latent topic information to obtain a more discriminative representation of these hard data points. In this way, a more effective pairwise-based ranking classifier can be obtained, taking into account hidden topics that are automatically detected as part of the training process. After presenting the details of our model, we discuss a technique for solving the resulting optimization problem. Finally, we conduct extensive experiments on several benchmark datasets and show that our proposed model consistently improves the state-of-the-art approaches. We also present results related to the query-wise performance of our model in comparison to the comparative models and show that our model performs much better.

2. RELATED WORK

2.1 Learning-to-rank models

Maximum margin learning has already been successfully applied to LTR in a number of different ways [1, 14], in both pointwise [22] and pairwise [7] LTR models. For example, the latter approach uses `RankSVM` with a pairwise hinge loss function which is specifically adapted to the LTR task. Note that while [5] discusses the use of query topics in a maximum margin setting, it does not rely on a topic model for inferring these query topics. A large number of other learning-to-rank models have recently been proposed. For example, Gao et al. [13] presented a novel semi-supervised listwise LTR model to deal with domains for which no training data is available. A sparse learning-to-rank model for information retrieval has been proposed in [17]. Finally, Dang et al. [9] proposed a two-stage LTR framework to improve the performance in cases where many relevant documents are excluded from the ranking list by bag-of-words retrieval models. None of the above works consider latent topics for tackling the LTR problem.

2.2 Latent topic models

Unsupervised topic models such as Latent Dirichlet Allocation (`LDA`) [6] have proven very effective for ad-hoc information retrieval [30, 32, 10]. The usefulness of topic models in this context stems from the fact that latent topics can be used to better estimate the similarity between queries and documents when the overlap in actual content words is low [18, 12]. Existing approaches do not follow the LTR paradigm, but rather use the low-dimensional topic space to conduct traditional document retrieval under an unsupervised setting, without considering any other features.

Supervised models based on a latent semantic space have also been considered [18]. In [4], the authors have proposed a discriminative model, called supervised semantic indexing, which can compute query-document and document-document similarity in a semantic space. While the authors state that their model can easily be extended to an LTR setting, they have not developed such an extension. Gao et al. have proposed topic models which jointly consider the query and the title of a document, with the aim of improving document retrieval in a language modeling framework [12, 15]. Even though they also use posterior regularization, there are major differences with our approach; for instance, our model is designed for the LTR task. In addition, we introduce a novel interpretation to the optimization framework of the posterior distribution obtained using `LDA`, using a convex optimization technique, which leads to a novel formulation and inference algorithm.

There is another line of work that uses probabilistic topic modeling for document classification, which takes into account labeled training data during parameter estimation. One example is the supervised topic model from [21], which introduces a response variable in the topic modeling framework. The maximum margin entropy discrimination model, known as `MedLDA` [34], discriminatively learns a topic model for binary and multi-class document classification. A difference between our work and such maximum margin supervised topic models is that our model is designed for solving LTR tasks whereas the aforementioned models only consider document classification, which leads to a different optimization problem.

Note that both our model and the models proposed in [34, 35] are learned by solving a regularized Bayesian inference task. However, `MedLDA` and infinite latent `SVM` solve a different optimization problem. In particular, these models use the same input as text classification models, and use a latent linear discriminant function with a random weight vector that only encapsulates the topic feature weights, leading to an expected classifier with an effective discriminant function. Due to the use of a random weight vector, these models cannot directly take into account the non-random pre-computed features that are traditionally considered in the LTR task. Supervised text classification models also face problems when dealing with unbalanced data [22], where one class tends to dominate the training set, as is commonly the case for the class of non-relevant documents in information retrieval. Such unbalanced data leads to a classifier where the minority class is totally ignored by the text classification model [22]. Therefore, novel methodologies and algorithms are needed in order to solve the LTR task using topic models with pairwise maximum margin constraints.

2.3 Regularized posterior inference

Some probabilistic models have been proposed which make use of posterior inference with regularization, although latent topics have not previously been considered in such models. For example, in [11, 29] the authors proposed a probabilistic regularization framework for structured weakly supervised learning. They showed that by directly imposing decomposable regularization on the posterior moments of latent variables, the computational efficiency of the unconstrained model can be retained while ensuring that desired constraints hold in expectation. Supervised topic models discussed above such as [34, 16] also conduct posterior regularization.

3. BACKGROUND

3.1 Problem Definition

The learning-to-rank (LTR) paradigm aims to learn an optimal ranking function for a given set of available relevance features. Specifically, let a query set Q and document collection D be given, and assume that each query q_u from Q is associated with r_u documents. The learning algorithm is given x queries as training examples. The corresponding documents are typically retrieved from a search engine and will be denoted as $D_u = \{d_1^u, d_2^u, \cdots, d_{r_u}^u \in D\}$. Furthermore assume that in the training data, a discrete relevance label h_i^u is associated with each document d_i^u. Our objective is to learn a ranking function from this training data that can be used to rank documents for previously unseen queries.

As is typical in LTR settings, we assume that we have access to a mapping $\psi : Q \times D \rightarrow \mathbb{R}^n$ that maps each query-document pair to an n-dimensional feature vector. The training algorithm is then given labeled examples of the form $T = (T_1, T_2, \cdots, T_x)$, with each T_u of the form $T_u = ((\psi_1^u, h_1^u), (\psi_2^u, h_2^u), \cdots, (\psi_{r_u}^u, h_{r_u}^u))$ and $\psi_i^u = \psi(q_u, d_i^u)$. We then need to learn a real-valued function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ such that $f(\psi(q, d_i)) > f(\psi(q, d_k))$ if document d_i is more relevant to query q than document d_k.

3.2 Pairwise Maximum Margin LTR

We will learn a linear ranking function, parametrized by a weight vector $\boldsymbol{\eta} \in \mathbb{R}^n$:

$$f_{\boldsymbol{\eta}}(\psi) = \boldsymbol{\eta}^{\intercal}.\psi, \tag{1}$$

Our framework follows the pairwise approach to LTR. In other words, we treat LTR as a binary classification problem by constructing for each pair of documents (d_i^u, d_k^u), associated with a given query q_u, the vector $(\psi_i^u - \psi_k^u)$. The latter vector is assigned the class label y_{ik}^u defined as follows:

$$y_{ik}^u = \begin{cases} +1 & \text{if } d_i^u \text{ is ranked higher than } d_k^u \\ -1 & \text{otherwise.} \end{cases} \tag{2}$$

Let the set B_u of document index pairs, given a query q_u, be defined as follows:

$$B_u = \{(i,k) | h_i^u > h_k^u\} \tag{3}$$

In other words, $(i,k) \in B_u$ if d_i^u is more relevant than d_k^u.

Our framework is based on a maximum margin classifier model, similar to Support Vector Machines (SVM), which requires us to solve the following optimization problem:

$$\begin{aligned} \underset{\boldsymbol{\eta}}{\text{minimize}} \quad & \frac{1}{2}||\boldsymbol{\eta}||^2 + C \sum \xi_{ik} \\ \text{subject to} \quad & \xi_{ik} \geq 0 \\ & y_{ik}^u \boldsymbol{\eta}^{\intercal}(\psi_i^u - \psi_k^u) \geq 1 - \xi_{ik}, \forall u, i, k, \end{aligned} \tag{4}$$

where C is a regularization parameter and ξ_{ik} are the non-negative slack variables. It can be shown that solving this optimization problem is equivalent to minimizing the empirical hinge loss function, defined as follows:

$$L_h(B_u) = \sum_{(i,k)\in B_u} (1 - y_{ik}^u \boldsymbol{\eta}^{\intercal}(\psi_i^u - \psi_k^u)). \tag{5}$$

In our framework, we will consider a modified hinge loss function, which takes into account the difference in relevance degree and the different number of document pairs for different queries:

$$L'_h(B_u) = \frac{1}{|B_u|} \sum_{(i,k)\in B_u} [|h_i^u - h_k^u| - y_{ik}^u \boldsymbol{\eta}^{\intercal}(\psi_i^u - \psi_k^u)]. \tag{6}$$

Advantages of considering a query-level loss function have been discussed in [25, 1], where the relevance levels of different documents were also taken into account.

3.3 Topic Modeling

The topic modeling component of our framework is based on Latent Dirichlet Allocation (LDA). The aim of LDA is to represent the meaning of each document as a probability distribution over a set of latent topics. It assumes a generative process, in which each word in a document is generated by sampling a topic and then sampling a word. LDA assigns each word in the document collection to one of the latent topics (initially at random), and uses this assignment to estimate both the probability distribution over topics associated with each document and the probability distribution over words associated with each topics. These probability distributions are then used to improve the topic assignments of the words, and the whole process is repeated until convergence.

To apply LDA in our setting, we expand the document collection D with the set of queries from the training data Q, i.e. queries are treated as short documents to construct an aggregated document collection, denoted as $\mathbb{D}$. For any document d in $\mathbb{D}$, we let N^d denote the number of words in document d. Furthermore, we let $\boldsymbol{w}^d = \{w_n^d\}_{n=1}^{N^d}$ denote the words appearing in the each document and $\boldsymbol{W} = \{\boldsymbol{w}^d\}_{d=1}^{\mathbb{D}}$ the words in the entire aggregated document set. Let $\boldsymbol{z}^d = \{z_n^d\}_{n=1}^{N^d}$ denote the topic assignment of the words in document d and let $\boldsymbol{Z} = \{\boldsymbol{z}^d\}_{d=1}^{\mathbb{D}}$ denote the topic assignment of all words in $\mathbb{D}$. Let $\boldsymbol{\Theta} = \{\theta^d\}_{d=1}^{\mathbb{D}}$ be the topic distributions of all documents in $\mathbb{D}$. Let the number of topics be K. Let $\boldsymbol{\Phi} = \{\phi_1, \phi_2, \cdots, \phi_K\}$ be the $V \times K$ matrix of topic distribution parameters, where each ϕ_k parameterizes a topic-specific multinomial word distribution. V denotes the number of words in the vocabulary. The posterior distribution is then given by:

$$P(\boldsymbol{\Theta}, \boldsymbol{Z}, \boldsymbol{\Phi} | \boldsymbol{W}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{P_0(\boldsymbol{\Theta}, \boldsymbol{Z}, \boldsymbol{\Phi} | \boldsymbol{\alpha}, \boldsymbol{\beta}) P(\boldsymbol{W} | \boldsymbol{\Theta}, \boldsymbol{Z}, \boldsymbol{\Phi})}{P(\boldsymbol{W} | \boldsymbol{\alpha}, \boldsymbol{\beta})}, \tag{7}$$

where $P_0(\boldsymbol{\Theta}, \boldsymbol{Z}, \boldsymbol{\Phi} | \boldsymbol{\alpha}, \boldsymbol{\beta})$ is the prior probability, with $\boldsymbol{\alpha}$ the parameter of the Dirichlet prior on the per-document topic distributions and $\boldsymbol{\beta}$ the parameter of the Dirichlet prior on the per-topic word distributions.

Zellner in [33] has extended Bayes' rule so that it can be used as a learning model. Specifically, Zellner has showed that Bayes' rule can be transformed into an optimization problem. In this way, it can be shown that the values for $\boldsymbol{\Theta}$, $\boldsymbol{Z}$ and $\boldsymbol{\Phi}$ which maximize the posterior distribution (7) can be found by solving the following optimization problem:

$$\begin{aligned} \underset{P(\boldsymbol{\Theta}, \boldsymbol{Z}, \boldsymbol{\Phi} | \boldsymbol{\alpha}, \boldsymbol{\beta}) \in \mathbb{P}}{\text{minimize}} \quad & \mathrm{KL}[P(\boldsymbol{\Theta}, \boldsymbol{Z}, \boldsymbol{\Phi} | \boldsymbol{W}, \boldsymbol{\alpha}, \boldsymbol{\beta}) || P_0(\boldsymbol{\Theta}, \boldsymbol{Z}, \boldsymbol{\Phi} | \boldsymbol{\alpha}, \boldsymbol{\beta})] \\ & - \mathbb{E}_P[\log \mathrm{P}(\boldsymbol{W} | \boldsymbol{\Theta}, \boldsymbol{Z}, \boldsymbol{\Phi})] \\ \text{subject to} \quad & P(\boldsymbol{\Theta}, \boldsymbol{Z}, \boldsymbol{\Phi} | \boldsymbol{W}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \in \mathbb{P}, \end{aligned} \tag{8}$$

where $\mathbb{P}$ is the probability distribution space, $\mathrm{KL}(P||P_0)$ is the Kullback-Leibler divergence from P to P_0, and $\mathbb{E}$ is the expected value operator. This interpretation of Bayes' theorem will be useful for designing our pairwise LTR model, which will be based on an extension of (8). Note that $P(\boldsymbol{W} | \boldsymbol{\alpha}, \boldsymbol{\beta})$ has been omitted because it does not depend on $\boldsymbol{\Theta}, \boldsymbol{Z}, \boldsymbol{\Phi}$.

4. DESCRIPTION OF OUR FRAMEWORK

The key characteristic of our framework is that maximum margin learning is tightly integrated with topic discovery. To this end, in Section 4.1, we extend the optimization problem from (4) with a latent topic model, similar to Latent Dirichlet Allocation (LDA). As we discuss in Section 4.2, solving the resulting optimization problem requires us to alternate between topic discovery (informed by the parameters of the maximum margin classifier) and maximum margin label prediction (informed by the latent topic structure).

4.1 Unified LTR Model

The optimization view of Zellner's interpretation of Bayes' rule in (8) can be extended to incorporate posterior constraints in Bayesian inference, by adding a convex function to (8). This leads to the introduction of some auxiliary free parameters, which can be slack variables. We will use the following formulation:

$$\begin{aligned} \underset{P(\boldsymbol{\Theta},\boldsymbol{Z},\boldsymbol{\Phi}|\boldsymbol{\alpha},\boldsymbol{\beta})\in\mathbb{P},\boldsymbol{\xi}}{\text{minimize}} \; & \mathrm{KL}[P(\boldsymbol{\Theta},\boldsymbol{Z},\boldsymbol{\Phi}|\boldsymbol{W},\boldsymbol{\alpha},\boldsymbol{\beta})||P_0(\boldsymbol{\Theta},\boldsymbol{Z},\boldsymbol{\Phi}|\boldsymbol{\alpha},\boldsymbol{\beta})] \\ & - \mathbb{E}_P[\log \mathrm{P}(\boldsymbol{W}|\boldsymbol{\Theta},\boldsymbol{Z},\boldsymbol{\Phi})] + M(\boldsymbol{\xi}) \\ \text{subject to } & P(\boldsymbol{\Theta},\boldsymbol{Z},\boldsymbol{\Phi}|\boldsymbol{\alpha},\boldsymbol{\beta}) \in \mathbb{P}_{\text{new}}(\boldsymbol{\xi}), \\ & \xi \geq 0 \end{aligned} \tag{9}$$

where $\mathbb{P}_{\text{new}}(\boldsymbol{\xi})$ is the subspace of probability distributions satisfying the constraints that arise out of the optimization framework. The convex function $M(\boldsymbol{\xi})$ in our case will be based on the relevance-weighted pairwise query-level hinge loss defined in (6). Note that many other interesting extensions to Zellner's equation have been proposed in the past such as the MedLDA model [34].

To consider latent topics during maximum margin learning, the loss function in (6) needs to take into account the topic similarity between the query q_u and document d_i^u. If we denote this similarity by Υ_i^u, we can consider the following discriminant function:

$$\boldsymbol{\eta}^{\mathsf{T}}(\psi_i^u - \psi_k^u) + \eta_t(\Upsilon_i^u - \Upsilon_k^u). \tag{10}$$

where η_t is a parameter encoding the relative importance of the topic similarity Υ_i^u, which is calculated based on the topic document distribution $\boldsymbol{\Theta}$. Recall that in this distribution, a document (or query) d is represented as a $K \times 1$ vector $\vec{d_i^u}$, encoding for each latent topic z the corresponding probability $P(z|d)$. For document d_i^u associated with a query q_u, we define Υ_i^u as the cosine similarity between $\vec{d_i^u}$ and $\vec{q}_u$. Other metrics such as KL-divergence could also be used. The overall optimization problem we end up with is as follows:

$$\begin{aligned} \underset{\substack{P(\boldsymbol{\Theta},\boldsymbol{Z},\boldsymbol{\Phi}|\boldsymbol{\alpha},\boldsymbol{\beta}),\\ \boldsymbol{\eta},\eta_t,\boldsymbol{\xi}}}{\text{minimize}} \; & \mathrm{KL}[P(\boldsymbol{\Theta},\boldsymbol{Z},\boldsymbol{\Phi}|\boldsymbol{W},\boldsymbol{\alpha},\boldsymbol{\beta})||P_0(\boldsymbol{\Theta},\boldsymbol{Z},\boldsymbol{\Phi}|\boldsymbol{\alpha},\boldsymbol{\beta})] \\ & - \mathbb{E}_P[\log P(\boldsymbol{W}|\boldsymbol{\Theta},\boldsymbol{Z},\boldsymbol{\Phi})] + \frac{1}{2}(||\boldsymbol{\eta}||^2 + \eta_t^2) \\ & + \frac{C}{x}\sum_{u=1}^{x}\frac{1}{|B_u|}\sum_{(i,k)\in B_u}\xi_{(i,k)}^u \\ \text{subject to } & \xi_{(i,k)}^u \geq |h_i^u - h_k^u| - y_{ik}^u\big[\boldsymbol{\eta}^{\mathsf{T}}(\psi_i^u - \psi_k^u) + \\ & \eta_t(\Upsilon_i^u - \Upsilon_k^u)\big], \;\; \forall u,i,k, \\ & P(\boldsymbol{\Theta},\boldsymbol{Z},\boldsymbol{\Phi}|\boldsymbol{\alpha},\boldsymbol{\beta}) \in \mathbb{P}(\xi_{(i,k)}^u) \\ & \xi_{(i,k)}^u \geq 0, \;\; \forall u,i,k. \end{aligned} \tag{11}$$

Note that the latent topic similarities Υ_i^u and Υ_k^u are computed in the regularized topic space arising from the optimization component, which differentiates our approach from the two-stage heuristic methods described in Section 1. By directly regularizing the posterior distribution with the maximum margin constraint, we obtain a more powerful model, catered specifically to the pairwise LTR task. The usefulness of this approach stems from the fact that latent topics can be chosen specifically to help the maximum margin classifier, e.g. by preventing instances from being located near the margin.

4.2 Solving the Optimization Problem

As solving (11) exactly is intractable, we resort to a Monte Carlo method which alternates between two steps. In the first step, our goal is to find a maximum margin separation of the points, i.e. we determine $\boldsymbol{\eta}$ and η_t given $P(\boldsymbol{\Theta},\boldsymbol{Z},\boldsymbol{\Phi}|\boldsymbol{W},\boldsymbol{\alpha},\boldsymbol{\beta})$. As in the existing RankSVM algorithm, the optimum solution can be found by adopting the Lagrangian method. In the second step, we estimate $P(\boldsymbol{\Theta},\boldsymbol{Z},\boldsymbol{\Phi}|\boldsymbol{W},\boldsymbol{\alpha},\boldsymbol{\beta})$ given $(\boldsymbol{\eta},\eta_t)$, as explained below. Both steps are repeated for a given number of iterations or until the sampler converges to a steady state. As in traditional topic modeling, the procedure starts with a random initialization of the topic assignments. Note that this mechanism closely resembles the Expectation-Maximization (EM) algorithm. However, while EM maximizes the expected log-likelihood under the marginal distribution of the latent variables, we are minimizing the regularized loss.

To estimate $P(\boldsymbol{\Theta},\boldsymbol{Z},\boldsymbol{\Phi}|\boldsymbol{W},\boldsymbol{\alpha},\boldsymbol{\beta})$, we use collapsed Gibbs sampling, where our goal is to estimate the model parameters $\boldsymbol{\Theta}$ and $\boldsymbol{\Phi}$. We present a brief derivation of the collapsed Gibbs sampler with maximum margin constraints below. Let n_{zw} denote the number of times a word w is assigned to a topic z and let p_{dz} denote the number of times a word from document d has been assigned to topic z. We can start with the joint distribution of the model, which can be expressed as:

$$P(\boldsymbol{Z}|\boldsymbol{\alpha}) \propto P(\boldsymbol{W},\boldsymbol{Z}|\boldsymbol{\alpha},\boldsymbol{\beta}) \tag{12}$$

When we incorporate the maximum margin framework in (12), we obtain the following model:

$$P(\boldsymbol{Z}|\boldsymbol{\alpha}) \propto \frac{P(\boldsymbol{W},\boldsymbol{Z}|\boldsymbol{\alpha},\boldsymbol{\beta})}{\Omega} \cdot \mathrm{e}^{\frac{1}{2}\Delta - \Psi}. \tag{13}$$

where Ω is the normalization constant. The first factor in the right-hand side of (13) adopts the collapsed Gibbs sampling formulation. In this sampling scheme, we also compute the transition probabilities, which are used to iteratively find the word-topic and document-topic latent topic distributions. The second factor corresponds to the regularization effects of the pairwise maximum margin classifier. Let Γ denote the Gamma function where $\Gamma(x) = (x-1)!$. By integrating out $\boldsymbol{\Theta}$ and $\boldsymbol{\Phi}$, we get the marginalized posterior distribution:

$$\begin{aligned} P(\boldsymbol{Z}|\boldsymbol{\alpha}) \propto & \prod_{z=1}^{K}\left(\frac{\Gamma(\sum_{s=1}^{|\boldsymbol{\beta}|}\beta_s)}{\prod_{s=1}^{|\beta|}\Gamma(\beta_s)}\int\prod_{w=1}^{V}\phi_{zw}^{n_{zw}+\beta_w-1}\mathrm{d}\boldsymbol{\phi}_z\right)\cdot \\ & \prod_{d=1}^{\mathbb{D}}\left(\frac{\Gamma(\sum_{s=1}^{|\boldsymbol{\alpha}|}\alpha_s)}{\prod_{s=1}^{|\boldsymbol{\alpha}|}\Gamma(\alpha_s)}\int\prod_{z=1}^{K}\theta^{p_{dz}+\alpha_z-1}\mathrm{d}\boldsymbol{\theta}_d\cdot\mathrm{e}^{\frac{1}{2}\Delta-\Psi}\right) \end{aligned} \tag{14}$$

where Δ and Ψ are defined as follows

$$\Delta = \Bigg[\sum_{u=1}^{x} \sum_{(d,k)\in B_u} \sum_{\hat{u}=1}^{x} \sum_{(\hat{d}\hat{k})\in B_{\hat{u}}} \lambda_{dk}^{u} \lambda_{\hat{d}\hat{k}}^{\hat{u}} \cdot [(\psi_d^u - \psi_k^u) + (\Upsilon_d^u - \Upsilon_k^u)]\Bigg] \cdot [(\psi_{\hat{d}}^{\hat{u}} - \psi_{\hat{k}}^{\hat{u}}) + (\Upsilon_{\hat{d}}^{\hat{u}} - \Upsilon_{\hat{k}}^{\hat{u}})] \tag{15}$$

$$\Psi = \sum_{u=1}^{x} \sum_{(d,k)\in B_u} \lambda_{dk}^{u} (h_d^u - h_k^u) \tag{16}$$

with λ_{dk}^{u} and $\lambda_{\hat{d}\hat{k}}^{\hat{u}}$ the Lagrange multipliers, satisfying $0 \leq \lambda_{dk}^{u} \leq \frac{C}{x \times |B_u|}$ for all d, k and u. The above formulation can be used to derive the following updating rule for the Gibbs sampler, for one variable z_n^d given the others:

$$P(z_n^d = t | \boldsymbol{Z}_{\neg n}, w = v, \boldsymbol{W}_{\neg n}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \tag{17}$$
$$\propto \left(\frac{n_{tv} + \beta_{w_n^d} - 1}{\big[\sum_{v=1}^{V} n_{tv} + \beta_v\big] - 1} \right) (p_{dt} + \alpha_t - 1) \mathrm{e}^{\frac{1}{2}\Delta - \Psi}$$

In the above equation, the current word is excluded in the counts n_{zw} and p_{dz} which we denote with the symbol $\neg$. The parameters in the model can be estimated based on the following formulations:

$$\phi_{zv} = \frac{n_{zv} + \beta_v}{\left(\sum_{v=1}^{V} n_{zv} + \beta_v \right)} \tag{18}$$

$$\theta_{dz} = \frac{p_{dz} + \alpha_z}{\left(\sum_{z=1}^{K} p_{dz} + \alpha_z \right)} \mathrm{e}^{\frac{1}{2}\Delta - \Psi} \tag{19}$$

4.3 Prediction on Unlabeled Data

To apply our LTR model on unseen data, we have to determine the latent dimensions of the unseen data using the regularized topic space that was learned from the training data. In addition, the model has to project the unseen data into the learned ranking space of the pairwise maximum margin classifier for label prediction. To this end, we use the point estimate of the topics computation procedure from the training data. After the Markov chain has reached a certain number of iterations, we draw J samples from it. Specifically, using the maximum a posteriori probability (MAP) estimation scheme, we obtain a new set of topic distributions $\boldsymbol{\Phi}$, which we write as $\hat{\boldsymbol{\Phi}}$. Using our collapsed Gibbs sampler, an estimate of $\hat{\boldsymbol{\Phi}}$ can be obtained as follows:

$$\hat{\phi}_{zw} \propto \frac{1}{J} \sum_{j=1}^{J} n_{zw}^{(j)} + \beta_w \tag{20}$$

The latent dimensions of an unseen document π can then be computed as follows:

$$P(z_n^{\pi} = t | \boldsymbol{Z}_{\neg n}) \propto \hat{\phi}_{t w_n^{\pi}} (p_{\pi t} + \alpha_t) \tag{21}$$

The above formulations can be used to separately compute the latent dimensions for the query and the documents. Let $\boldsymbol{\pi}$ denote a feature vector for π, containing the same features as those considered in the training set. The prediction formula for the pairwise maximum margin classifier can be expressed as:

$$F(\boldsymbol{\pi}) = \sum_{u=1}^{x} \sum_{(i,k)\in B_u} \lambda_{ik}^{u} [(\psi_i^u - \psi_k^u) + (\Upsilon_i^u - \Upsilon_k^u)].\boldsymbol{\pi} \tag{22}$$

In the right-hand side of (22), the expression $\sum_{u=1}^{x} \sum_{(i,k)\in B_u} \lambda_{ik}^{u} [(\psi_i^u - \psi_k^u) + (\Upsilon_i^u - \Upsilon_k^u)]$ computes the feature weights for the new vector $\boldsymbol{\pi}$. The recipe is to alternately run, until convergence is reached, the topic prediction model depicted in (21) using the parameters of the trained regularized topic space, and then use the pairwise maximum margin formulations depicted in (22), which provides the pairwise regularization effect to the latent dimensions computed in the previous step, for predicting the labels. When (21) and (22) are combined together following the same paradigm as depicted in (13), the new documents are "folded-in" in the previously trained regularized topic space.

5. EXPERIMENTS AND RESULTS

5.1 Datasets

Many benchmark LTR collections have been released in the past, such as LETOR [24] and the Yahoo! LTR Challenge dataset. These collections contain pre-computed query-dependent and query-independent features but do not provide access to the corresponding documents, which means that the latent topic features cannot be computed. As a result, such existing public benchmark LTR collections cannot be used in our experiments. One exception is the LETOR OHSUMED [24] benchmark collection, whose text and queries are freely available[1]. In addition to LETOR OHSUMED, we will use three new collections, which we have built based on well-known TREC datasets. First, we have used AQUAINT, which is used in the TREC-HARD [2] track and contains a total of 50 queries[2]. The queries along with the corresponding annotations are provided on the TREC HARD disk. Second, we have used the WT2G dataset[3], consisting of 50 TREC queries[4]. Finally, we have used the ClueWeb09 Category B English documents collection. In particular, we used a list of 91 features from [3], where the authors have considered 150 TREC Web Track queries from 2009 to 2011.

Recently, Terrier v4.0 has introduced LTR in its distribution[5]. We used this open source search engine to create training, testing and validation sets for our two new datasets (AQUAINT and WT2G). We created the corresponding LTR datasets using all query-dependent and query-independent models currently available in Terrier v4.0. Note that Terrier v4.0 uses the "title" field from the TREC topics file to retrieve documents. The created datasets comprise 39 normalized features, which form a subset of the features described in [20]; all considered features fall in the WM and WMP classes described in [20]. Terrier v4.0 can be set to use its default `BM25` implementation to retrieve the top 1000 documents, which corresponds to what is used in the official LETOR [24] collections. We separately created the LTR dataset for ClueWeb09 because of two reasons. First, this dataset is extremely large for Terrier to handle on a standalone machine. Second, Terrier's implementation currently cannot compute the 91 considered features for the ClueWeb09 dataset. In all text collections,

[1] http://ir.ohsu.edu/ohsumed/

[2] Note that we use the term queries here whereas in TREC usually the term topics is used.

[3] http://ir.dcs.gla.ac.uk/test_collections/access_to_data.html

[4] http://trec.nist.gov/data/t8.web.html

[5] http://terrier.org/docs/current/learning.html

we used stemming (Porter stemmer) and stopword removal, as implemented in Terrier v4.0. Subsequently, we created five folds. Each fold has approx. 60% query-document pairs for training, approx. 20% query-document pairs for testing and the rest for validation. Note that the training, testing and validation sets do not share any queries. One can thus notice that we have followed the standard LTR dataset creation procedure as reported in the LETOR dataset documentation [24]. In our experiments, 42681 unique documents were retrieved by Terrier for the 50 queries in the AQUAINT collection. For the WT2G dataset, we retrieved 40773 unique documents, whereas the ClueWeb09 dataset consists of 42044 unique documents. Note that our implementation of topic modeling considers the complete documents, irrespective of different fields (e.g. TITLE) that may be present.

The publicly available LETOR OHSUMED dataset contains training (approx. 60% of query-document pairs), testing (approx. 20% of query-document pairs) and validation (approx. 20% of query-document pairs) splits in five folds. It consists of 14430 unique documents. More information about this dataset can be obtained from [1]. Each feature vector in the LETOR OHSUMED also contains the document identifier. This information can help us relate a document from the downloaded OHSUMED text documents with its corresponding feature vector in LETOR OHSUMED. LETOR OHSUMED also contains meta-information which allows us to relate text queries with their corresponding query identifiers.

5.2 Comparative Methods

Many popular models have been implemented in RankLib[6]: `Listnet`, `AdaRank-MAP`, `AdaRank-NDCG`, `Coordinate Ascent`, `LambdaMART` and `MART`, which belong to the listwise class of LTR models, as well as the pairwise models `RankNet`, `RankBoost` and `LambdaRank`, and the pointwise model `Linear Regression - L2 Norm` and `Random Forests`. Note that models implemented in RankLib have been used as strong state-of-the-art baselines in many recent works such as [26, 23]. We will compare our method against the aforementioned methods, as well as a number of other publicly available implementations: `SVMMAP`[7], a listwise model, `RankSVM-Struct`[8], a pairwise model, and the recently proposed `DirectRank`.

Finally, we will also compare our approach with a semantic search model similar to the one proposed in [4], which we have slightly modified to handle the feature instances needed for the LTR task; we will refer to this model as `Semantic Search`. Note that while the authors of [4] mention that their model can be adapted to incorporate LTR features, they have not included such an evaluation in their work. This model bears some resemblance to our proposed framework, in that both models are capable of generating latent topic similarity features dynamically. In particular, `Semantic Search` can be made to generate topic similarity with minor changes to the original model. However, a crucial difference is that we compute topic similarity in the regularized space, whereas `Semantic Search` computes topic similarity in a so-called "concept space", which is very similar to the space obtained by the latent semantic indexing (`LSI`) model. We have also experimented with a variant of `Semantic Search`, proposed in [12], but found its performance on LTR datasets to be very similar to, but slightly weaker than that of the model from [4]. Finally, note that we have not considered any unsupervised topic models for comparison, as such models cannot make use of relevance judgements during the training process, and can therefore not be competitive with LTR models in this context.

5.3 Experimental Setup

Hyperparameter Values: We use the following symmetric hyperparameter values in the `LDA` topic model: $\alpha = \frac{50}{K}$ and $\beta = 0.01$, where K is the number of topics. This value is used throughout all experiments, for both our model and other approaches that use latent topics. These values have also been used e.g. in `LDA` for document retrieval [30]. We have experimented with different hyperparameter values, but did not obtain any significantly different results. We used the validation set for model selection, which is common for LETOR baselines [26, 19]. All NDCG results have been averaged across five folds.

Parameter tuning for comparative models: The parameters of the comparative LTR models in each fold were tuned using the validation set. To replicate the results for LETOR OHSUMED, we followed the official guidelines[9], which state: "*The validation set can only be used for model selection (setting hyper-parameters and model structure), but cannot be used for learning. Most baselines released in LETOR website use MAP on the validation set for model selection; you are encouraged to use the same strategy and should indicate if you use a different one.*" For consistency, we used the same rules for model selection in the three other datasets. Parameters were tuned for each fold separately, as is usual for the LTR setting.

Parameter tuning for our model: We need to tune the regularization parameter C, the learning rate and the number of topics K for each fold. In each fold, we varied C from 0.1 to 1 in steps of 0.1, and K from 10 to 1000 in steps of 10. The learning rate values that we experimented with are $\{10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}\}$. The value of K was varied from 10 to 200 in steps of 10. The number of iterations of the Markov chain was set to 1000. Using these ranges for the parameters, as for the comparative models, we searched for the configuration that produced the best mean average precision (MAP) score on the validation set. Note that because we use regularized Bayesian inference, overfitting in our model is unlikely, despite using 1000 iterations for the Markov chain. Moreover, there is evidence that models based on Bayesian inference in general tend not to overfit [28].

Description of the experiments: We have conducted two types of experiments. First, we have compared our approach with the comparative models, using their default feature sets, as described in Section 5.1. The results of this experiment are discussed in Section 5.4.1. While this allows us to find out whether our approach can outperform the existing state-of-the-art LTR models, it should be noted that we are using a latent topic similarity feature, which other models do not have access to, and it might be that our model is performing better simply because it has access to better features. Therefore, in a second experiment, we have added

[6] http://people.cs.umass.edu/∼vdang/ranklib.html

[7] http://projects.yisongyue.com/svmmap/

[8] http://research.microsoft.com/en-us/um/beijing/projects/letor/letor4.0/baselines/ranksvm-struct.html

[9] http://research.microsoft.com/en-us/um/beijing/projects/letor//letor3baseline.aspx

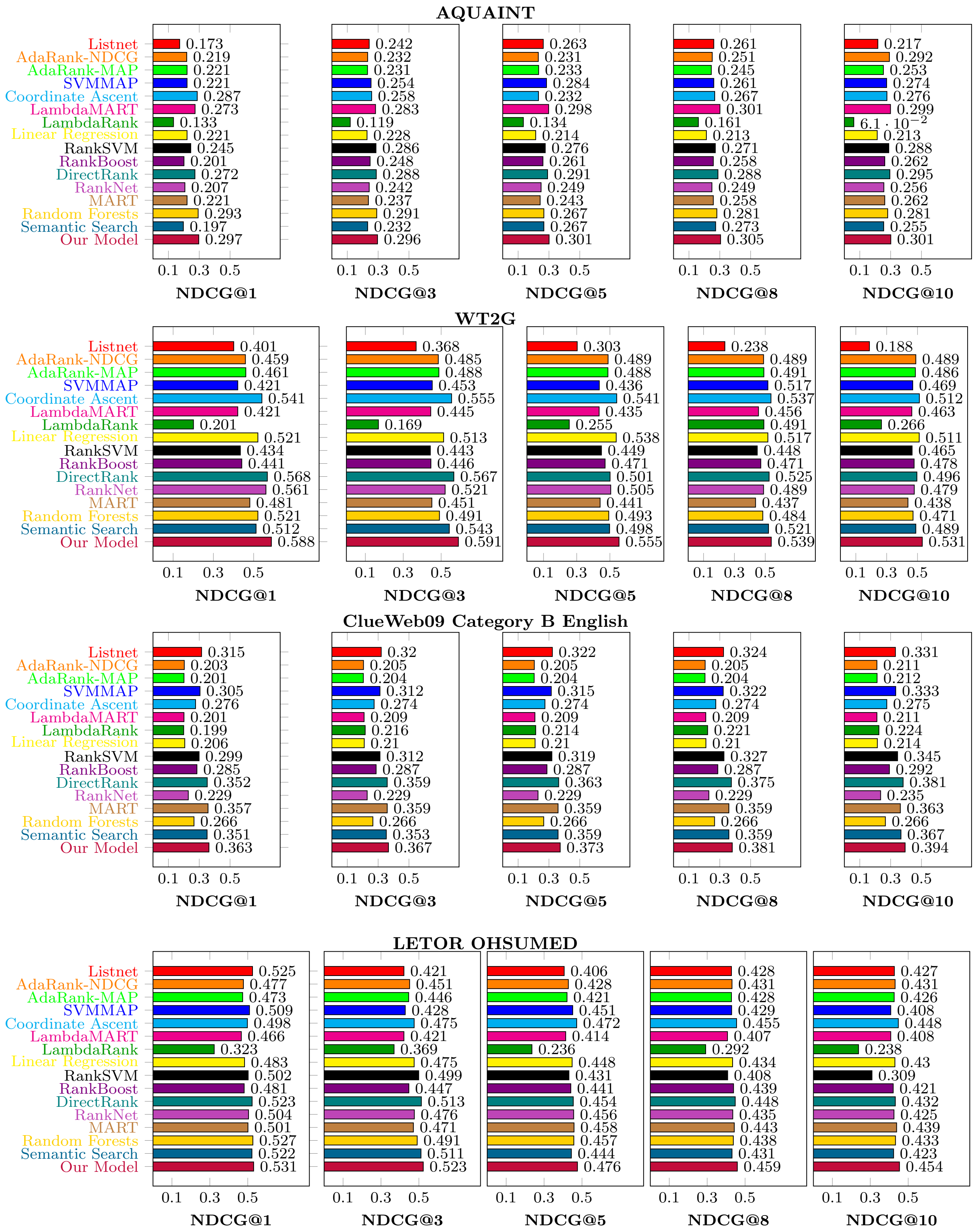


Figure 1: LTR results for four datasets.

the query-document latent topic similarity as an additional feature for the comparative models. In particular, we have used `LDA`, following the procedure described in [30] to independently compute query-document topic similarity. We also experimented with the cosine similarity for comparing queries and documents in the latent topic space, but the results were inferior to those of the method from [30]. As for our model, the number of latent topics K was automatically determined using the validation set by varying the number of topics from 10 to 200 in steps of 10. The number of iterations in the Gibbs sampler for computing the topic similarity using the method from [30] is 1000. This experiment will allow us to assess whether our integrated model has any benefits over models that simply use latent topics as one of the features. The results are presented in Section 5.4.2.

Evaluation metric: We have used the PERL evaluation tool available in LETOR 3.0 for evaluating all the models[10]. Our evaluation metric is NDCG which is popular for LTR evaluation; we refer to [1] for a definition of this metric. We will use NDCG@k to present our main results, where $k = 1, 3, 5, 8, 10$. We have tested for statistical significance using the paired t-test, as is usual for LTR experiments.

5.4 Experimental Results

5.4.1 Traditional LTR Comparison

Figure 1 presents the results for the considered datasets. As can be seen from the figure, our model consistently outperforms all the comparative models, for each of the datasets and NDCG cut-offs. The improvements against each of the comparative methods are statistically significant according to the paired t-test with $p < 0.05$. For the AQUAINT dataset, `DirectRank`, `Semantic Search`, `Coordinate Ascent`, and `LambdaMART` do much better than the other comparative models. For the WT2G dataset, `LambdaRank` performs very poorly, with our model being the only one that clearly differs. The results for ClueWeb09 and LETOR OHSUMED are consistent with the results for AQUAINT. In both cases, our model considerably outperforms the other, while `DirectRank` also shows good performance among the comparative models. These results confirm our intuition that integrating latent topic information with maximum margin learning is capable of outperforming the state-of-the-art approaches.

To better understand the differences in performance across different queries, Table 1 shows a winning number comparison. We refer to [19] for a detailed description of this metric, which has also been used in e.g. [26]. As we can see from the table, our model substantially outperforms all of the comparative models, with `DirectRank` again the best performer among the comparative methods.

5.4.2 Topic Enhanced Datasets Experiment

We present the results for the second experiment in Figure 2. As we can see from this figure, the performance of many comparative methods has slightly improved or has remained same as in the previous result. For example, in the AQUAINT dataset, performance of many comparative models has improved and none of the results have deteriorated. Therefore, in the majority of cases latent topic information leads to a slightly improved performance. However, even with the latent topic feature added, none of the comparative methods can outperform our approach. The improvement is still statistically significant for all datasets and all models, according to the paired t-test with $p < 0.05$, except at NDCG@8 in WT2G dataset where `Coordinate Ascent` model performs equally well. For the LETOR OHSUMED dataset, our model performs strictly better than each of the comparative methods, except at NDCG@8 where `Coordinate Ascent` performs equally well. `Linear Regression` model also shows good performance in this experiment. We can see that the performance gap between the comparative models and our model has slightly decreased compared to Figure 1. Table 2 presents the winning number results. Comparing this table with Table 1 clearly shows that the relative performance of the comparative methods has improved. This demonstrates that latent topic information is helpful, even when used as a standard feature in other LTR models. However, our approach still performs consistently better, confirming the benefit of using an integrated model.

Models	N@1	N@3	N@5	N@8	N@10
`ListNet`	17	16	13	13	14
`AdaRank-NDCG`	12	14	16	17	18
`AdaRank-MAP`	15	12	14	11	12
`SVMMAP`	15	20	18	20	18
`Coordinate Ascent`	24	25	26	25	27
`LambdaMART`	21	23	21	19	22
`LambdaRank`	05	08	06	02	11
`Linear Regression`	18	19	15	18	20
`RankSVM`	18	15	19	15	15
`RankBoost`	16	12	13	18	15
`DirectRank`	30	34	31	32	34
`RankNet`	19	21	19	12	15
`MART`	18	18	22	16	15
`Random Forests`	25	20	21	20	18
`Semantic Search`	28	33	25	28	25
`Our Model`	**33**	**39**	**37**	**36**	**36**

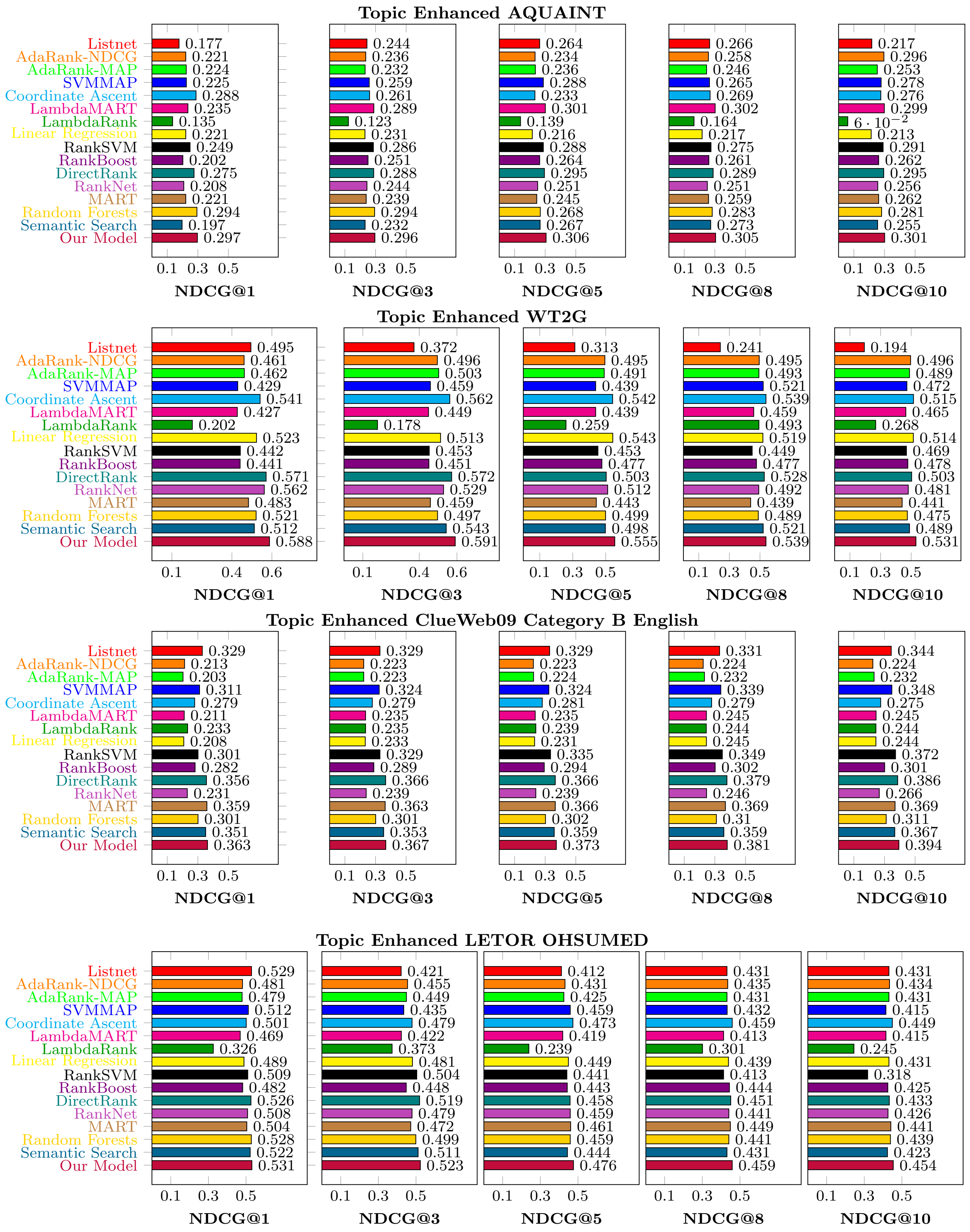
Table 1: Winning number comparison.

Models	N@1	N@3	N@5	N@8	N@10
`ListNet`	19	17	14	16	18
`AdaRank-NDCG`	13	16	17	19	20
`AdaRank-MAP`	16	14	15	10	15
`SVMMAP`	16	21	20	19	20
`Coordinate Ascent`	27	26	24	20	23
`LambdaMART`	23	27	20	21	23
`LambdaRank`	02	04	03	05	12
`Linear Regression`	19	20	18	20	21
`RankSVM`	20	17	20	18	16
`RankBoost`	18	15	15	19	17
`DirectRank`	28	28	29	25	25
`RankNet`	19	18	21	18	19
`MART`	19	19	24	19	18
`Random Forests`	23	19	24	21	19
`Semantic Search`	25	31	26	23	22
`Our Model`	**30**	**33**	**31**	**28**	**27**

Table 2: Winning number comparison when the latent topic similarity feature is incorporated in the comparative methods.

When looking in more detail at the per-query performance in each dataset, we note that our model especially outperforms the comparative methods for long queries, e.g. consisting of two words or more. In Table 3, we present this comparison in terms of the percentage of winning numbers,

[10] http://research.microsoft.com/en-us/um/beijing/projects/letor/letor3download.aspx



Topic Enhanced AQUAINT

Topic Enhanced WT2G

Topic Enhanced ClueWeb09 Category B English

Topic Enhanced LETOR OHSUMED

Figure 2: LTR results in topic enhanced datasets.

	1	2	3	>3
AQUAINT	54.33	59.74	69.12	78.66
WT2G	59.64	67.35	73.92	73.92
ClueWeb	61.23	72.98	79.76	83.21
LETOR OHSUMED	57.34	63.65	69.44	75.61

Table 3: Query level performance in terms of the percentage of winning numbers for different query lengths.

which shows that the relative performance also improves with longer queries. These comparisons are done by considering the latent topic feature in the comparative methods. This result is expected, as longer queries make it easier to assign meaningful topics to the query.

6. CONCLUSION

We have presented a novel LTR model that combines latent topic information with maximum margin learning in a unified way. In our model the latent topic representation is directly regularized with a pairwise maximum margin constraint, which leads to more informative latent topics. We have conducted extensive experiments using benchmark collections and have shown clear improvements over the state-of-the-art comparative methods. The main strength of our model stems from the fact that topic similarity is computed in the regularized latent topic space. This allows for a direct interplay between the pairwise maximum margin classifier and the topic model, which ensures that the topics which are learned are maximally informative for the prediction of labels. As our experiments show, such an integrated approach outperforms the existing two-stage de-coupled approach to incorporating latent topics for in LTR models. One interesting line for future work would be to look at different LTR tasks such as temporal LTR, for which the use of temporal topic models seems promising.

7. REFERENCES

[1] S. Agarwal and M. Collins. Maximum margin ranking algorithms for information retrieval. In *ECIR*, pages 332–343. 2010.

[2] J. Allan. HARD track overview in TREC 2003 high accuracy retrieval from documents. Technical report, DTIC Document, 2005.

[3] N. Asadi and J. Lin. Effectiveness/efficiency tradeoffs for candidate generation in multi-stage retrieval architectures. In *SIGIR*, pages 997–1000, 2013.

[4] B. Bai, J. Weston, D. Grangier, R. Collobert, K. Sadamasa, Y. Qi, O. Chapelle, and K. Weinberger. Learning to rank with (a lot of) word features. *IR*, 13(3):291–314, 2010.

[5] J. Bian, X. Li, F. Li, Z. Zheng, and H. Zha. Ranking specialization for web search: A divide-and-conquer approach by using topical RankSVM. In *WWW*, pages 131–140. ACM, 2010.

[6] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *JMLR*, 3:993–1022, 2003.

[7] Y. Cao, J. Xu, T.-Y. Liu, H. Li, Y. Huang, and H.-W. Hon. Adapting ranking SVM to document retrieval. In *SIGIR*, pages 186–193, 2006.

[8] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, and H. Li. Learning to rank: from pairwise approach to listwise approach. In *ICML*, pages 129–136, 2007.

[9] V. Dang, M. Bendersky, and W. B. Croft. Two-stage learning to rank for Information Retrieval. In *ECIR*, pages 423–434. 2013.

[10] D. Duan, Y. Li, R. Li, R. Zhang, and A. Wen. RankTopic: Ranking based topic modeling. In *ICDM*, pages 211–220, 2012.

[11] K. Ganchev, J. Graça, J. Gillenwater, and B. Taskar. Posterior regularization for structured latent variable models. *JMLR*, 11:2001–2049, 2010.

[12] J. Gao, K. Toutanova, and W.-t. Yih. Clickthrough-based latent semantic models for Web search. In *SIGIR*, pages 675–684, 2011.

[13] W. Gao and P. Yang. Democracy is good for ranking: Towards multi-view rank learning and adaptation in web search. In *WSDM*, pages 63–72, 2014.

[14] R. Herbrich, T. Graepel, and K. Obermayer. Large margin rank boundaries for ordinal regression. *NIPS*, pages 115–132, 1999.

[15] J. Jagarlamudi and J. Gao. Modeling click-through based word-pairs for Web search. In *SIGIR*, pages 483–492, 2013.

[16] Q. Jiang, J. Zhu, M. Sun, and E. P. Xing. Monte Carlo methods for maximum margin supervised topic models. In *NIPS*, pages 1592–1600, 2012.

[17] H. Lai, Y. Pan, C. Liu, L. Lin, and J. Wu. Sparse learning-to-rank via an efficient primal-dual algorithm. *IEEE Transactions on Computers,*, 62(6):1221–1233, 2013.

[18] H. Li and J. Xu. Semantic matching in search. *FTIR*, 8, 2013.

[19] T.-Y. Liu. *Learning to rank for Information Retrieval.* Springer, 2011.

[20] C. Macdonald, R. L. Santos, and I. Ounis. The whens and hows of learning to rank for Web search. *IR*, 16(5):584–628, 2013.

[21] J. D. Mcauliffe and D. M. Blei. Supervised topic models. In *NIPS*, pages 121–128, 2008.

[22] R. Nallapati. Discriminative models for Information Retrieval. In *SIGIR*, pages 64–71, 2004.

[23] S. Niu, Y. Lan, J. Guo, X. Cheng, and X. Geng. What makes data robust: A data analysis in learning to rank. In *SIGIR*, pages 1191–1194, 2014.

[24] T. Qin, T.-Y. Liu, J. Xu, and H. Li. LETOR: A benchmark collection for research on learning to rank for Information Retrieval. *IR*, 13(4):346–374, 2010.

[25] T. Qin, X.-D. Zhang, M.-F. Tsai, D.-S. Wang, T.-Y. Liu, and H. Li. Query-level loss functions for Information Retrieval. *IPM*, 44(2):838–855, 2008.

[26] M. Tan, T. Xia, L. Guo, and S. Wang. Direct optimization of ranking measures for learning to rank models. In *KDD*, pages 856–864, 2013.

[27] J. Tang, N. Liu, J. Yan, Y. Shen, S. Guo, B. Gao, S. Yan, and M. Zhang. Learning to rank audience for behavioral targeting in display ads. In *CIKM*, pages 605–610, 2011.

[28] Y. W. Teh. Dirichlet process. In *Encyclopedia of Machine Learning*, pages 280–287. 2010.

[29] K. Vorontsov and A. Potapenko. Additive regularization of topic models. *ML*, pages 1–21, 2014.

[30] X. Wei and W. B. Croft. LDA-based document models for ad-hoc retrieval. In *SIGIR*, pages 178–185, 2006.

[31] F. Xia, T.-Y. Liu, J. Wang, W. Zhang, and H. Li. Listwise approach to learning to rank: Theory and algorithm. In *ICML*, pages 1192–1199, 2008.

[32] X. Yi and J. Allan. Evaluating topic models for Information Retrieval. In *CIKM*, pages 1431–1432, 2008.

[33] A. Zellner. Optimal information processing and Bayes's theorem. *The American Statistician*, 42(4):278–280, 1988.

[34] J. Zhu, A. Ahmed, and E. P. Xing. MedLDA: maximum margin supervised topic models. *JMLR*, 13(1):2237–2278, 2012.

[35] J. Zhu, N. Chen, and E. P. Xing. Bayesian inference with posterior regularization and applications to infinite latent SVMs. *JMLR*, 15:1799–1847, 2014.