A Unified Optimization Framework for Robust Pseudo-relevance Feedback Algorithms

Joshua V. Dillon[*]
College of Computing
Georgia Institute of Technology
jvdillon@gatech.edu

Kevyn Collins-Thompson
Microsoft Research
1 Microsoft Way
Redmond, WA 98052 U.S.A.
kevynct@microsoft.com

ABSTRACT

We present a flexible new optimization framework for finding effective, reliable pseudo-relevance feedback models that unifies existing complementary approaches in a principled way. The result is an algorithmic approach that not only brings together different benefits of previous methods, such as parameter self-tuning and risk reduction from term dependency modeling, but also allows a rich new space of model search strategies to be investigated. We compare the effectiveness of a unified algorithm to existing methods by examining iterative performance and risk-reward tradeoffs. We also discuss extensions for generating new algorithms within our framework.

Categories and Subject Descriptors: H.3.3 [**Information Retrieval**]: Retrieval Models
General Terms: Algorithms, Experimentation
Keywords: Query expansion, optimization

1 Introduction

A relatively new advance in IR is the development of *risk-aware* algorithms that not only attempt to perform well on average across queries, but which seek to dynamically adjust their behavior from query to query to reduce their *variance* or instability – especially to avoid serious errors. As one example of such a task, it is well known that the effectiveness of pseudo-relevance feedback can be highly sensitive to a number of parameters, such as the number of terms, or number of top-ranked documents chosen. Thus, robust algorithms seek to reduce instability by finding reliable values for these parameters automatically, removing the need to commit to a single operational setting for all queries.

In one example, Collins-Thompson [4] introduced a novel view of query expansion as a portfolio optimization problem, resulting in a constrained quadratic program (which we call the CT algorithm) that finds a reliable, effective set of feedback terms by combining term covariance information with a set of task-specific constraints that prune out bad expansion models. Their approach operates as a post-process on a set of candidate terms and assumes little about the underlying retrieval model. It operates only in the space of expansion terms and and may return the *empty set* of expansion terms if expansion is deemed too risky for a particular query.

On the other hand, Tao and Zhai [20] introduced a robust pseudo-relevance model (which we call the TZ algorithm) based on the language modeling approach that jointly solves for both term and document weights. Unlike the CT algorithm, it includes document weights in the model, but does not model term dependencies or have the ability to prune a sparse subset of expansion terms. While the CT objective is convex and uses 'hard' constraints, the TZ algorithm uses a non-convex likelihood objective, finding local maxima with regularized EM, with a soft, successively relaxed initial penalty on models far from the initial query.

In this work we unify these two seemingly unrelated approaches in a principled way to produce a pseudo-relevance feedback algorithm that jointly determines both the optimal *term subset* and the optimal *document subset* to use for a given query, while also allowing a rich set of new potential constraints and improved objective structure, so that term and document dependencies, sparsity, and so on, is easily added. Our evaluation includes standard evaluation metrics, iterative analysis, and a parameter space visualization computed on a high-performance computing cluster showing regimes of mean and variance of performance attained across parameter sweeps.

Making progress on the robust pseudo-relevance feedback problem is important not only for potentially improved result quality, but also because increasingly available context data in Web search engines need a principled framework for exploiting them to model the underlying information need. In addition, improving a query representation has other applications, such as broad matching of search advertisements with web pages. Finally, pseudo-relevance feedback can be seen as an instance of a broader feature selection problem under uncertainty, so better techniques for pseudo-relevance feedback may lead to better, more generally applicable feature selection methods in other areas of information retrieval or machine learning that must deal with limited, noisy training examples and uncertainty in parameter estimation.

[*]Work partially conducted during an internship at Microsoft Research.

CIKM'10, October 26–30, 2010, Toronto, Ontario, Canada.

2 Optimization framework

We derive our optimization model in three steps. First, we describe a generative model introduced by Tao and Zhai that conditions queries and documents on latent variables. Second, we give some background on the TZ algorithm for estimating the parameters of the model using EM and a regularized maximum likelihood objective. Third, we show how to merge the non-convex likelihood of the Tao-Zhai (TZ) algorithm with the convex risk-reward optimization of the Collins-Thompson (CT) algorithm in a principled way by invoking a general optimization family called a Convex-Concave (CCCP) program. Finally, we discuss how our CCCP analysis gives an easy way to generate new algorithms that fix limitations of the TZ algorithm while also allowing new domain-specific constraints and objectives. The overall result is an extremely flexible optimization framework for constructing effective searches for high-quality query representations that can account for problem structure.

2.1 A basic generative model of queries and documents

```
GENERATE-RANDOM-DOCUMENT($L, \alpha; \theta_T, \theta_B$)
$c \leftarrow$ ZEROS$(1, K)$ // observed rv
$z \leftarrow$ ZEROS$(1, K)$ // latent rv
for $j \leftarrow 1$ to $L$
do $f \leftarrow$ FLIP-HEADS-BIASED-COIN$(\alpha)$
    if $f$ is HEADS
       then $w \leftarrow$ ROLL-BIASED-DIE$(\theta_T)$
               $z[w] \leftarrow z[w] + 1$
       else $w \leftarrow$ ROLL-BIASED-DIE$(\theta_B)$
    $c[w] \leftarrow c[w] + 1$
return $c$
```

Figure 1: The Tao-Zhai (TZ) generative procedure for a single document in the feedback set. This model posits documents are a two-part mixture of multinomials (on a per-word basis). L denotes the length of the document, i.e., number of multinomial samplings, and α its relevant/nonrelevant mixture. Documents are assumed independent, given the parameters.

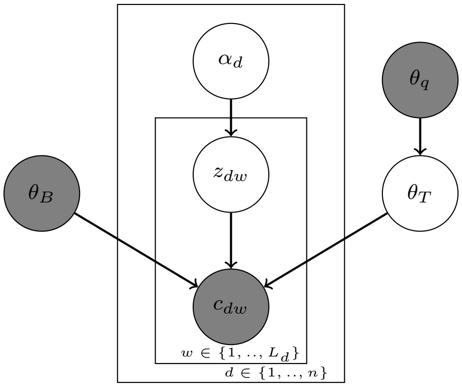


Figure 2: Graphical model depiction of (above) generative procedure (for n documents). Nodes are random variables, with shaded nodes being observed (user's query θ_q, fixed corpus background model θ_B, word counts c_{dw}). Plates indicate conditionally independent replications.

C_{dw}	Multinomial random variable indicating number of occurrences of term w in doc d and instantiated as c_{dw} (lower case). The complete matrix is C.
Z_{dw}	Binomial random variable indicating term w of doc d is relevant to query and instantiated as z_{dw} (lower case). The complete matrix is Z.
θ_T	Parameters for K-dimensional topic model T (implicitly dependent on q) which correspond to C.
α_d	Document relevance parameters (F-dimensional) which correspond to Z_d.
$F_k(q)$	Top-k feedback documents associated with q from corpus $\mathcal{C}$, $F_k(q) \subseteq \mathcal{C}$; F for brevity.
V	Vocabulary; with a slight abuse in notation, words are assumed to have values in $V = \{1, 2, \ldots, V\}$.
θ_q	Multinomial parameters for query q, i.e. $\theta_{qw} = c_{qw} / \sum_w c_{qw}$.
θ_B	Multinomial parameters for background model B, ie, $\theta_{Bw} = \sum_{d \in \mathcal{C}} c_{dw} / \sum_{d \in \mathcal{C}} \sum_w c_{qw}$.
Δ	Vector of parameters, typically $\Delta = [\alpha\ \theta_T]$.
μ	Effective sampling size for query, i.e. $\theta_T \sim \mathsf{Dir}(1+\mu\theta_q)$.

Figure 3: Guide to notation used in this paper.

Tao & Zhai [20] introduced the simple generative model and optimization algorithm for pseudo-relevance feedback[1] shown in Figures 1 and 2, in which feedback documents $F_k(q)$ are generated from a mixture of two multinomial language models: a background model θ_B and a 'relevant topic' model θ_T. We assume these models use a vocabulary $\mathcal{V}$ of dimension K. The TZ model assumes that the topic model θ_T has a fixed Dirichlet prior, $\mathsf{Dir}(1 + \mu\theta_q)$. One distinctive feature of their model is that it jointly optimizes both query and document weights simultaneously: the feedback documents are endowed with 'relevance' weights α_d which are to be learned jointly with the word probabilities of θ_T.

From this assumed generative algorithm, we note that the complete-data log-likelihood (for a single document) is

$$\log \Pr(c_d, z_d|\Delta) = \sum_{w=1}^{V} \big[z_{dw} \log\big(\theta_{Tw}\alpha_d\big) + (c_{dw} - z_{dw}) \log\big(\theta_{Bw}(1 - \alpha_d)\big)\big] \tag{1}$$

with model prior,

$$\log \Pr(\Delta|\mu, \theta_q) \propto \log \Pr(\theta_T|\mu, \theta_q) = \mu \sum_{w=1}^{V} \theta_{qw} \log \theta_{Tw}. \tag{2}$$

where we have disregarded all normalization terms as they are constant in Δ.

We note the similarity of this model to LDA, viz., a two topic LDA with one topic held fixed, and non-informative or constant priors. Since this is a two-topic model, the conditional $\Pr(z_{dw}|c_{dw}, \Delta)$ is binomial, leaving the incomplete-data log-likelihood as a multinomial with parameters $\alpha\theta_T + (1 - \alpha)\theta_B$. Learning this model is made difficult by the fact that this log-sum is not linear.

[1] Our notation mostly follows that used in Tao & Zhai, with some extensions. Omitting the subscript indicates the matrix/vector, as opposed to the specific element, e.g. α is a vector of α_i and $[\theta_B]_w = \theta_{Bw}$. Also, the value of a variable, e.g. α, after k iterations is denoted by $\alpha^{(k)}$.

Several approaches exist to cope with this problem. Tao & Zhai employ standard EM manipulations which yield the following two-step iterative procedure:
E-step:

$$z_{dw}^{(k)} \leftarrow c_{dw} \frac{\alpha_d^{(k)} \theta_{Tw}^{(k)}}{\theta_{Tw}^{(k)} \alpha_d^{(k)} + \theta_{Bw}^{(k)} (1 - \alpha_d^{(k)})} \tag{3}$$

M-step:

$$\alpha_d^{(k+1)} \leftarrow \frac{\sum_{j=1}^{V} z_{dj}^{(k)}}{\sum_{j=1}^{V} c_{dj}} \tag{4}$$

$$\theta_{Tw}^{(k+1)} \leftarrow \frac{1}{\lambda} \left(\mu \theta_{qw} + \sum_{i=1}^{n} z_{iw}^{(k)} \right) \tag{5}$$

with Lagrangian λ the appropriate normalization of $\theta_{Tw}^{(k+1)}$.

Typically the E- and M-steps are repeated until convergence, however TZ deviate from this practice by imposing a schedule of decay on parameter μ, i.e., $\mu \leftarrow \mu\delta^k$ where $\delta \in (0, 1)$. Convergence is heuristically defined by the crossover point between the current value of μ and the expected number of relevant words in the feedback set using the model at iteration k. By forcing the training procedure to pursue only the most significant parameter updates, this approach is intended to cope with high term variance without incurring additional modeling overhead. It plays a role somewhat analogous to a prior and helps to prevent overfitting.

However, this approach lacks a certain flexibility. The ability to access fine control over the query prior is inextricably coupled to the rate of convergence of the EM algorithm. This convergence is in turn tied to complicated term-term covariances and document length variability. By reformulating the updates in such a way that accounts for the *rate of change* in updates, we obtain both a clearer interpretation of μ as well as finer control across iterations and within (by accounting for parameter covariation).

2.2 Rewriting the TZ algorithm as a Convex-Concave Program

The above EM algorithm seeks a parametrization that maximizes the posterior likelihood under the query-driven Dirichlet prior. Here, we seek an alternative procedure based on the same generative model, but in which we may gain additional flexibility: first by exploring alterative regularizations to improve stability, and secondly by formulating additional objectives or constraints in *the space of latent variables*, instead of parameter space. The motivation for the latter is that task-specific knowledge can often be expressed by functions of expectations over the latent variables that have an intuitive interpretation and give more control over the parameter estimation process. For example, if we have feedback observations about the utility of specific words or documents, or even specific words in specific documents, we can easily incorporate these as constraints in the latent variable space. To do this, we study generalizations of EM in which the basic closed-form E-step on latent variables is replaced by a more general convex optimization problem.

2.2.1 Background

The general idea of EM is to increase the incomplete-data likelihood $\ell(\Delta; C)$ through maximizing some function of the complete-data likelihood; it is assumed that this object, $\ell(\Delta; C, Z)$, is easier to manipulate. Such a formulation can easily be understood through developing a lower bound using the information theoretic functionals, cross-entropy $H(p, q)$, entropy $H(p) = H(p, p)$, and KL-divergence $D(p||q) = H(p, q) - H(p)$. For details, see chapter 2 of [7].

$$\begin{aligned} \ell(\Delta; C) &= \sum_{i=1}^{n} \log p_\Delta(X^{(i)}, Z^{(i)}) \\ &= \sum_{i=1}^{n} \sum_{z \in \mathcal{Z}} q(z) \log \left(\frac{p_\Delta(X^{(i)}, z}{p_\Delta(z|X^{(i)})} \right) + nH(q(Z)) \\ &= -\sum_{i=1}^{n} D\big(q(Z)||p_\Delta(X^{(i)}, Z)\big) + \sum_{i=1}^{n} D\big(q(Z)||p_\Delta(Z|X^{(i)})\big) \\ &\geq -\sum_{i=1}^{n} D\big(q(Z)||p_\Delta(X^{(i)}, Z)\big) \triangleq -\mathcal{F}_n(q, \Delta) \end{aligned} \tag{6}$$

The last inequality follows from the non-negativity of KL-divergence, a result commonly known as Gibbs' inequality. We note that this bound is valid for any distribution $q(Z)$.

Standard EM derivations typically choose

$$q(Z) \equiv \Pr(Z|C, \Delta') \tag{7}$$

to exploit the (assumed) simplicity of the conditional and the fact that the entropy portion of $\mathcal{F}$ becomes constant for Δ. Hence, maximizing (6) reduces to minimization of cross-entropies, $\sum_i H\big(\Pr_{\Delta'}(Z_i|C_i), \Pr_\Delta(C_i, Z_i)\big)$. By parameterizing q in this way, the search path taken toward the MLE is dictated by the precise characteristics of the complete-data model. If we were to allow the minimization to converge, this fact would be of lesser concern and it would be prudent to let computational efficiency constrain the choice of distribution family for q. However, since the TZ likelihood has an adaptive component, i.e., "decaying prior," the precise nature of the search path becomes more important. Accordingly, we aim to recast the optimization in a way that allows a larger space of potential distributions over Z, and thus, a richer set of search path strategies.

We begin by restating arguments of [16, 24] who note that an equivalent formulation of EM is,

$$\text{E-step:} \quad q^{(t+1)} \quad \leftarrow \underset{q}{\operatorname{argmin}} \{\mathcal{F}_n(q, \Delta^{(t)})\} \tag{8}$$

$$\text{M-step:} \quad \Delta^{(t+1)} \leftarrow \underset{\Delta}{\operatorname{argmin}} \{\mathcal{F}_n(q^{(t+1)}, \Delta) - \log \Pr(\Delta)\}. \tag{9}$$

Ignoring issues like local extrema, this formulation reveals that EM is equivalent to the joint minimizing of $\mathcal{F}_n(q, \Delta)$ with respect to q and Δ. Hence

$$q^* = \underset{q}{\operatorname{argmin}} \{\mathcal{F}_n(q, \Delta^*(q)) - \log \Pr(\Delta^*(q))\} \tag{10}$$

where

$$\Delta^*(q) = \underset{\Delta}{\operatorname{argmin}} \{\mathcal{F}_n(q, \Delta) - \log \Pr(\Delta)\} \tag{11}$$

also yields MLE solutions for the incomplete-data likelihood.

To simplify computation of (10) we treat all Z_{ij} as independent binomials with parameters (p_{ij}, c_{ij}). Although simple, this class is still a richer set of distributions than $\Pr(Z_{ij}|c_{ij}, \Delta)$. The entropy of $q(Z|p)$ conveniently decomposes into a term-count weighted combination of binary entropies, namely $H(q) = \sum_{i=1}^{n} \sum_{j=1}^{V} c_{ij} H(p_{ij})$ where p_{ij} represents the probability that word j of document i is relevant, i.e., $\Pr(Z_{ij}=1)$. Similar results apply to the cross-entropy term as well. The revised objective

$$p^* = \operatorname*{argmin}_p \{J(p) = u(p) - v(p)\}\,, \text{ where,}$$

$$u(p) = -\sum_{i=1}^{n}\sum_{j=1}^{V} c_{ij} H\big(p_{ij}\big) \tag{12}$$

$$v(p) = \log \Pr(\Delta^*(p)) + \sum_{i=1}^{n}\sum_{j=1}^{V} H\big(q(Z_{ij}|p_{ij}, c_{ij}), \Pr(Z_{ij}, c_{ij}|\Delta^*(p))\big), \tag{13}$$

enjoys several properties which we develop subsequently. Most notably, $J(p)$ is a difference of convex functions, i.e., negative entropy (u) plus the sum of a log-prior and negative cross-entropy as a function of $\Delta^*(q)$ (v) [24]. This is the final ingredient which allows us to recast the TZ likelihood into the more general optimization framework of Convex-Concave programs, which we now describe.

2.2.2 The Convex-Concave Procedure

Now that we have J written as the difference of convex functions u and v, we can minimize J using a generalization of EM known as the *Convex-Concave Procedure*. Yuille and Rangarajan [24] describe the Convex-Concave optimization procedure (CCCP) with the following recurrence:

$$x^{(k+1)} \leftarrow \operatorname*{argmin}_x \Big\{ u(x) - x^{\mathsf{T}} \nabla v(x^{(k)}) \Big\} \tag{14}$$

$$\text{such that } \begin{array}{l} c_i(x) \le 0, i \in \{1, \ldots, m\} \\ d_j(x) = 0, j \in \{1, \ldots, p\} \end{array}$$

where u, v, and c_i are real-valued convex functions, d_j is an affine function, v is differentiable, and all are defined on $\mathbb{R}^n$. More information on the convergence properties of the CCCP optimization family are given by Sriperumbudur and Lanckriet [18].

The advantage of the CCCP framework is that it is very general, and in fact includes all EM algorithms and some variational algorithms as special cases [24]. While other EM generalizations exist, CCCP also provides a recipe for deriving new algorithms for a very wide class of optimization problems, since almost any function can be expressed as a sum of convex and concave functions. Other techniques are not usually so broadly applicable.

The term $x^{\mathsf{T}} \nabla v(x^{(k)})$ is a linear approximation to v at $x^{(k)}$ and is known as a *majorizor* of $v(x)$ since it a tight upper bound for v at the point $x^{(k)}$. Substituting u from Eq. 12 and the derivative ∇v of Eq. 13 into Eq. 14 gives the convex optimization problem

$$p^{(k+1)} \leftarrow \operatorname*{argmin}_p \Big\{ -w^{\mathsf{T}} p - H(p) \Big\} \tag{15}$$

where $w = \nabla v(p^{(k)})$. This is an unconstrained maximum entropy problem that has a simple analytical solution, namely the matrix $\hat{p}$ with individual sigmoid entries

$$\hat{p}_{ij} = g_{ij}(W)/(1 + g_{ij}(W)) \tag{16}$$

where $g_{ij}(X) = \exp(-X_{ij}/C_{ij})$. Deriving the specific form of $w = \nabla v(\cdot)$ involves only basic calculus but due to space constraints we do not derive it here.

2.3 Unifying with the Collins-Thompson model

We now have a unifying view of the TZ and CT algorithms in one framework. Recall that the CT algorithm's objective[2] is a mean-variance tradeoff inspired by portfolio theory:

$$\hat{p} \leftarrow \operatorname*{argmin}_p \Big\{ -c^{\mathsf{T}} p + \frac{\kappa}{2} p^{\mathsf{T}} \Sigma p \Big\}. \tag{17}$$

Comparing this to the CCCP objective for the TZ likelihood in Eq. 15, both algorithms use the same bi-criterion form of objective with a linear function of p and a regularization function $R(p)$, which is an entropy term $R_{TZ}(p) = -H(p)$ for TZ and a quadratic term $R_{CT}(p) = x^{\mathsf{T}} \Sigma x$ for the CT algorithm. Both algorithms estimate expected values $p_{ij} \in [0,1]$ of latent variables: in the CT case p is a single $V \times 1$ vector, and in the TZ-CCCP case is a $V \times N$ matrix with one column for each document. Unlike TZ, the CT algorithm was run as a single-step post-process on an initial p_0 produced by a black-box feedback algorithm. Also, the CCCP prior term $\log \Pr(\Delta^*(p))$ in $v(p)$ (Eq. 13) can be seen as a soft constraint corresponding to the CT algorithm's hard query support constraint forcing query term p_i values to stay close to 1 in the solution.

In the default TZ model, interactions between terms in θ_T are not explicitly modeled, whereas the CT captures term dependencies using the matrix Σ in the quadratic term. The regularization term $R(p)$ in the CCCP objective, however, gives us a place to add dependencies between p_{ij}. We can either use a single Σ for all documents, or estimate a matrix Σ_d individually for each document. In the next section, we describe how translation kernels can be used to effectively estimate Σ.

2.4 Adding term dependency information via translation kernels

One effective method of estimating semantic term dependency is to define a statistical translation process between terms using *translation kernels* [9]. We create a translation kernel by first computing a similarity graph between all pairs of terms. For vertices u and v, the edge weight $e(u, v)$ is defined as a function of $f_u(w)$, the co-occurrence frequency of term u with term w in the top-ranked documents, giving a matrix E with entries

$$e_{uv} = exp(-\frac{1}{\sigma^2} \arccos^2 \sum_w \sqrt{f_u(w) f_v(w)}) \tag{18}$$

where the sum is taken over all words w in the vocabulary $\mathcal{V}$. The graph heat kernel is computed via the matrix exponential of the normalized graph Laplacian

$$\mathcal{L} = D^{-1/2}(D - E)D^{-1/2} \tag{19}$$

where D is a diagonal matrix with $D_{ii} = \sum_j e_{ij}$. The matrix exponential $H = exp(-t\mathcal{L})$ models the flow of heat across the graph as a function of time parameter t, which controls the amount of translation. For small t, $H \approx I$ and for large t, H is approximately uniform. Finally, we interpolate the submatrix T of Σ by computing $\hat{T} = (1 - \lambda)T + \lambda H$.

[2] In the CT model, p is a $1xV$ vector with entries $p_i = p(w_i|\theta_R)$ for each word w_i in vocabulary V and relevance model θ_R. Σ is a $V \times V$ positive definite term dependency matrix. The parameter κ specifies the mean-variance tradeoff. The weights c were derived from a Relevance Model estimated from top-ranked documents.

Note that the matrix Σ is applied at each step and thus affects the *relative* change in p. Thus, the role of a low weight in entry Σ_{vw} is to penalize or restrict changes to the model in directions where v and w covary. For example, terms that have high translation probability from the query will be treated almost as conservatively as the query itself: the optimization is reluctant to make large-magnitude changes away from not only query terms but closely related ones. Thus, we retain the conservative strategy of staying close to the initial query, but with the advantage of a flexible, more semantically rich definition of distance.

2.5 Further extensions

Rather than modify the CCCP objective function of Eq. 15 directly to add additional factors like term dependency into the objective, we can consider a modular approach that is also applicable to standard EM formulations: solve for $\hat{p}$ analytically (either from the CCCP step or the default EM E-step), and then find the closest latent variable matrix $X^{(k)}$ to $\hat{p}^{(k)}$ but subject to additional conditions. By 'closest', for this paper we use the Frobenius norm, a standard distance measure for matrices. We refer to this as the *constrained E-step* method. We now gives examples showing how new constraints and objectives can be added to this framework.

Constraints can play an important role in helping to eliminate low-quality models from consideration by encoding their properties (or rather, the opposite of them) to define a 'hard' feasible set of the problem. In many cases, linear constraints are sufficient to encode a rich variety of conditions. Note that because these are iterative algorithms, the constraints can in theory also be dynamic, changing with each iteration to reflect important local factors such as feature confidence. We now give two examples of how extensions can be fit into the constrained E-step framework.

Diversity constraints In the search for reliable solutions, we may consider that relying too heavily on only a small number of uncertain latent variables is too risky. Instead, we could implement a *diversity constraint* over the words in each document, so that no more than η_w-percent of the total probability mass can be allocated to the top r_w terms. It turns out that this can be expressed as a linear constraint ([1], p.279) via auxiliary vectors u_j of size $|V|$ and a scalar variable t_j for each document d_j, along the columns of $X^{(k)}$. This diversity constraint appears superficially similar to standard smoothing methods, in that it acts to redistribute probability mass from higher-probability events to lower-probability ones. However, unlike standard smoothing, relative changes in latent variable mass can change significantly from iteration to interation, due to the nature of the top-k criterion and hard upper-bound on the mass[3]. We also note that the *aspect balance constraint* over query models in the Collins-Thompson query expansion approach is a type of linear diversity constraint [4].

Term dependencies We can also make use of the translation kernels described in Section 2.4. As before, we create a translation kernel $\Sigma_T = exp(-t\mathcal{L})$ with time parameter t, which controls the amount of translation. We use a translation strength parameter λ to combine Σ_T with p using $(I + \lambda\Sigma_T)$.

$$\begin{aligned}
&\underset{X}{\operatorname{argmin}} && \|X - (I + \lambda\Sigma_T)\hat{p}\|_F && \textit{E-step dist.} && (20)\\
&\text{subject to} && \Sigma_i X_{ij} = \Sigma_i \hat{p}_{ij} && \textit{Doc mass invariant} && (21)\\
& && r_W \cdot t_j + \mathbf{1}^T u_j \leq \eta_W && \textit{Diversity constr I} && (22)\\
& && t_j + u_j \geq x_j / c_j && \textit{Diversity constr II} && (23)\\
& && c_j = \Sigma_{i=1}^{V} \hat{p}_{ij} && && (24)\\
& && u_j \geq 0 && && (25)\\
& && 0 \geq X \geq 1 && \textit{Label consistency} && (26)
\end{aligned}$$

Figure 4: The basic constrained E-step for finding the closest matrix X to the default E-step matrix $\hat{p}$, while respecting diversity constraints over a document's latent variables for words, and using a translation kernel Σ_T in the objective. Here, $j = 1 \ldots F$ over the set of F feedback documents.

An optimization step that brings all these together is shown in Figure 4. To use it in a standard EM algorithm, we simply replace the normal E-step with our convex program, and use the optimal solution matrix $\hat{X}$ instead of the default matrix $\hat{p}$. To use it with the CCCP program, we can apply it to the $\hat{p}$ solution from Eq. 15. In Section 4.5 we do a basic evaluation of the effects of these generalizations on the risk-reward tradeoff of the query model estimation algorithm. We leave further exploration of objectives and constraints for future work.

3 Related work

The most relevant previous studies are of the two algorithms by Collins-Thompson [3] and Tao & Zhai [20] described earlier. Xu and Akella [22] replaced the TZ two-mixture generative model with a Dirichlet Compound Multinomial, using a different latent variable model and closed-form E-step based on simulated annealing. It would be interesting to explore the use of the latter's more sophisticated generative model within our CCCP optimization framework.

CCCP and related algorithms have seen increased use recently for machine learning problems. For example, Yu and Joachims gave a CCCP for learning structural Support Vector Machines with latent variables [23]. CCCPs themselves are connected to a broad class of *majorization-minorization* algorithms, in which EM is a special case. Such frameworks have been introduced and motivated by problems in areas like image restoration [10], but we have not seen much application yet to information retrieval problems.

Previous work on regularization schemes can be divided into two types: term score smoothing, and document score smoothing. In the document score domain, Diaz [8] introduced the use of regularization that smoothed over the graph of document-document similarities. It would be interesting to investigate this type of smoothing in the parameter matrix of our model, in addition to the term-term smoothing that our translation model does. In the term domain, Mei and Zhai [15] described smoothing language models over graph structures. Unlike these previous approaches, our framework can model structure between terms and docu-

[3] We note that a similar form of diversity constraint could be applied to *documents* by constraining the rows of $X^{(k)}$, which hold the latent variables for occurrences of a single word w across all documents. We might prefer states in which there must be stronger evidence across multiple documents instead of relying on a single source. Variance-based Markowitz-type diversity [4] is another possibility. These are topics for future work.

ments, not just between entities of the same type. In motivating the importance of modeling term dependencies, we note a recent study by Udapa *et al.* [21] that confirmed the importance of accounting for term dependencies and set-level properties in finding higher-quality expansion sets, compared to searching for expansion terms individually.

The use of heat-transfer kernels for query expansion is another contribution of this paper. The closest previous work on random walk models for query expansion [6] also used a term dependency graph in which word co-occurrence was one of several dependency types, with combined transition edge weights estimated using logistic regression. The heat-transfer approach has the advantage of giving a more interpretable, statistically principled derivation of term transition probabilities.

Finally, we have drawn inspiration from a key reference work by Graca et al. [11], who proposed modifying EM using a constrained E-step in order to model posterior constraints. They also gave theoretical results that give a penalized maximum likelihood interpretations to their framework, and gave examples of several natural-language applications, including statistical translation.

4 Evaluation

Here we confirm the utility of the basic CCCP formulation and include preliminary analysis of the effect of term dependency and diversity constraints in latent variable space on performance.

We use two standard test collections: TREC 1&2 (topics 51-200, TREC disks 1&2) and Robust 2004 (topics 301–450 and 601–700, TREC disks 4&5). We chose these partly because topics 301–450 (the TREC678 topic set) overlap with those used in the TZ study, while also adding 100 new queries that typically are more challenging for query expansion. Also, the RIA workshop [12] made available an extensive failure analysis over the TREC678 topic set. Indexing and retrieval were performed using the Indri 2.8 system in the Lemur toolkit [14]. We used the title fields of the TREC topics and phrases were not used. We also did not use stopping or stemming since we believe this removes potentially valuable word evidence, and that a principled approach should be able to make stopword and stemming decisions automatically as part of the estimation process. Document scoring was performed using query likelihood with the top 1000 documents retrieved and using Dirichlet query smoothing with $\mu = 2000$.

We also compute a Relevance Model expansion baseline [13] by first selecting, using the top 50 ranked documents, the top 1000 terms based on their Ponte [17] log-odds score for use as the vocabulary space V for θ_T. The top 20 expansion terms based on their Relevance Model probability were then selected as expansion terms. Note that the TZ study performed expansion by computing an EM solution over a large vocabulary and then truncating at the top 100 expansion terms.

4.1 Comparison of iterative gains

We first give a basic comparison between the basic CCCP iterative algorithm of Eq. 15 and the TZ algorithm. Figure 5 compares the gains that the CCCP and TZ algorithms achieve on the TREC 1&2 and Robust 2004 topics and collections, for both Mean Average Precision (MAP) and Precision-at-20 (P20). Each curve captures the gain or loss, compared to the initial (unexpanded) query, of a particular topic model θ_T model computed at each iteration and used as the query model for retrieval. In general, the range of the performance statistics is in accord with previous studies on the same topics and collections (e.g., in [5]).

Both the TZ and CCCP algorithms achieved their peak performance at around 40 to 50 iterations for both MAP and P20. Not coincidentally, this is the point at which the influence of the query prior starts to disappear when $\mu = 30000$ and $\delta = 0.9$. The CCCP algorithm achieved significantly higher peak MAP gain of 41.4% for TREC 1& 2, compared to the TZ peak MAP of 31.4%. Both algorithms outperformed the Relevance Model baseline gain of 29.1%. The relative performance of the algorithms was the same for the Robust 2004 collection, with the CCCP peak MAP gain of 13.3% being slightly better than the TZ peak MAP gain of 12.9%. For comparison, the Relevance Model MAP gain was only 1.6% on this collection.

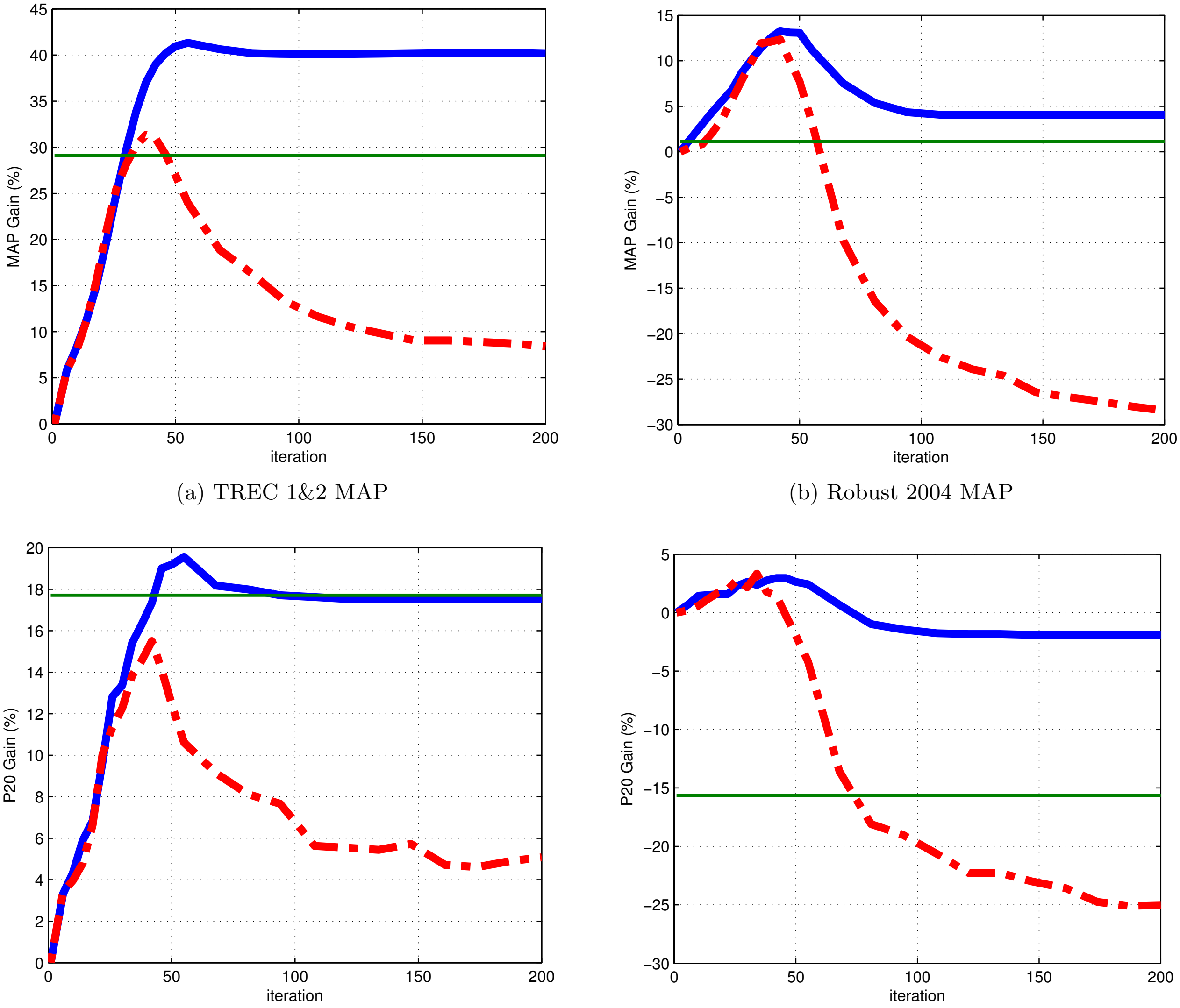
Beyond 50 iterations, the two algorithms behaved very differently. As the TZ algorithm ran toward convergence, it suffered from serious overfitting, resulting in rapidly deteriorating retrieval performance with each step. The CCCP algorithm, on the other hand, never experienced such a decline from the peak performance value and converged to a reasonable stable point within another 20 to 30 iterations. In practice, the TZ algorithm requires the use of an early stopping heuristic to avoid this overfitting problem, while the stability of the CCCP algorithm makes early stopping much less critical. Overall, the CCCP performance curves consistently dominated those of the TZ algorithm.

Table 1 summarizes performance using standard retrieval measures for the algorithms over both collections.

4.2 Sensitivity to initial parameters

The key parameters to be initialized are α_0, the starting weight of α_i for all documents, and μ_0 and δ for the query prior. The choice of α_0 can be seen as representing our initial belief in the likely quality of the feedback set for a typical query. Our experience with setting μ and δ matches that found by Tao & Zhai: as long as μ_0 is 'large' and δ is close to 1.0, the feedback performance is not much affected by varying those parameters.

The choice of α_0 had some effect on the overall performance of the CCCP algorithm, although peak improvements to MAP were strong across a fairly wide range of initial values. In particular, we found that a single operational setting of $\alpha_0 = 0.15$ worked well and gave close to optimal performance for both collections. Figure 6 shows the effect on Robust 2004 topics of varying α_0 for the CCCP algorithm, as a function of the number of iterations. For the curve with $\alpha_0 = 0.05$, the peak MAP value is 8.5%; this increases to 13.3% for $\alpha_0 = 0.10$, remains stable at 13.6% for $\alpha_0 = 0.15$, and then declines to 9.9% for $\alpha = 0.20$. In all cases, however, peak MAP occurs after approximately 50 iterations. In the long term, when the CCCP algorithm is allowed to run to convergence (roughly 100 iterations or more), the differences are more dramatic, ranging from a MAP gain of 6.2% at $\alpha_0 = 0.05\%$, to a small MAP loss -1.8% for $\alpha_0 = 0.20$. We also observed that the TZ algorithm was somewhat sensitive to the choice of α_0, within a much smaller operational range, i.e. between $1e^{-5}$ and $1e^{-7}$, with values closer to zero typically giving slightly better performance.



(a) TREC 1&2 MAP

(b) Robust 2004 MAP

(c) TREC 1&2 P20

(d) Robust 2004 P20

Figure 5: Comparison showing how the CCCP algorithm (thick solid line) achieves much more reliable gains (y-axis) compared to the TZ algorithm (dashed line) as iterations progress (x-axis). Shown are percentage gain/loss compared to using no expansion for MAP (top) and P20 (bottom) for TREC 1&2 and Robust 2004. In addition to being based on the same generative model, both TZ and CCCP use the same initial values of $\mu_0 = 30000$ and $\delta = 0.9$. The non-iterative Relevance Model expansion (Ponte-Lavrenko) is also shown for comparison (thin flat line).

Collection		NoExp	RM	TZ-FB	CCCP-FB
Robust 2004 (n=250)	MAP	19.91	20.23 (+1.61%)	22.48 (+12.91%)	22.56 (+13.31%)
	P5	41.76	41.92 (+0.38%)	42.64 (+2.11%)	42.40 (+1.53%)
	P20	30.52	30.62 (+0.33%)	31.64 (+3.67%)	31.42 (+2.95%)
TREC 1&2 (n=150)	MAP	15.62	20.17 (+29.13%)	20.52 (+31.37%)	22.08 (+41.36%)
	P5	39.73	39.73 (0.00%)	45.20 (+13.77%)	46.80 (+17.80%)
	P20	36.13	36.13 (0.00%)	41.73 (+15.50%)	43.20 (+19.57%)

Table 1: Comparison of MAP between Relevance Model (RM) baseline, Tao-Zhai (TZ-FB) and Unified (CCCP-FB) feedback methods. The best MAP for any iteration is shown. Precision improvement shown for all methods is relative to unexpanded query performance. (All numbers multiplied by 100.)

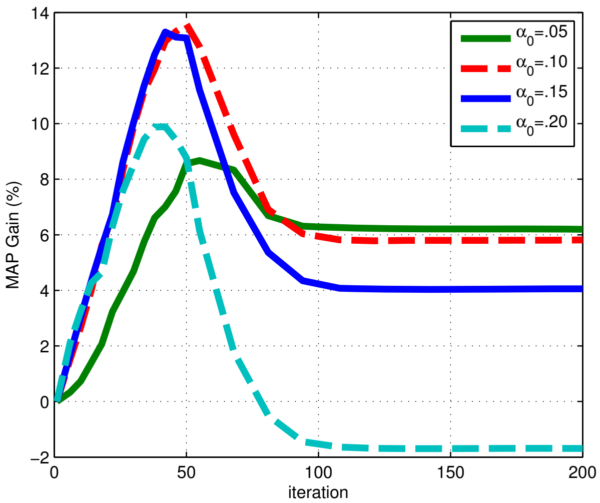


Figure 6: Sensitivity of CCCP algorithm performance to initial choice of α_0 for Robust 2004 topics.

TZ $p(w\|\theta_T)$	Word w	CCCP $p(w\|\theta_T)$	Word w
0.401	stirling	0.250	engine
0.096	cfc	0.207	stirling
0.087	kronor	0.047	cfc
0.053	134a	0.041	substitute
0.051	sdg	0.039	energy
0.049	chillers	0.036	containment
0.042	vattenfall	0.034	hfc
0.039	bleaching	0.033	sup
0.035	pfbc	0.024	hcfc
0.035	biofuels	0.023	pulp
AP 0.799		AP 0.979	

Figure 7: Sample query models for Tao-Zhai expansion (left) and CCCP method (right) for the TREC topic 447, 'stirling engine'. The top 10 expansion terms are shown for the optimal model found by each method.

4.3 Individual query analysis

We looked at the expansion terms and convergence of a particular query: TREC topic 447 'stirling engine'. We chose this topic because it was featured in the individual analysis of the original Tao-Zhai paper [20]. Note that our computed TZ term weights may be slightly different from those reported by [20]. We were careful to use the same EM parameters such as μ and δ. However, for efficiency the CCCP method works with a vocabulary of 100 candidate expansion terms. For this query, we experimented with limiting the TZ algorithm to the same 100-word candidate vocabulary, and performance dropped slightly to an AP of 0.3424, versus AP of 0.3549 for a full 50,000 word vocabulary.

After expansion, the TZ method obtained a maximum performance of AP 0.7989 after 28 iterations, with P5 of 1.0 and P20 of 0.60. Our CCCP unified method obtained an optimal AP of 0.9786 after 14 iterations, with P5 of 1.0 and P20 of 0.80. Both methods assign roughly equal total mass to the original query terms (49.6% for TZ vs 45.3% for CCCP), but assign it very differently: TZ gave most mass to the rarer term 'stirling' and had rapidly diminishing, almost sparse expansion weights, while the CCCP framework maintained a more even distribution over terms. We attribute this partly to the introduction of the maximum entropy term in the objective. For TREC query 312 ('hydroponics') also given in the Tao-Zhai paper, performance of TZ was slightly better, with an AP of 0.206 vs 0.1909 for CCCP.

4.4 Large-scale parameter space exploration

We also compared the TZ, CT, and hybrid algorithms in the space of achievable risk-reward tradeoffs [2] over a range of possible parameters, by using a massive parameter sweep on a large-scale computing cluster. For this experiment we used a proprietary internal Web corpus of 1.2 million documents and 400 queries with in-house relevance judgments. We used the Indri engine [19] in the Lemur toolkit [14] to retrieve the initial top-ranked documents for each query. Then for each query expansion algorithm, we sampled 5000 different combinations of parameters in that algorithm's high-dimensional 'box' of potential settings. Figure 8 shows the results as a risk-reward plot [3]. Here, the x-axis (risk) represents the percentage MAP loss averaged over queries that were hurt by expansion; the reward y-axis gives the percentage MAP gain/loss over *all* queries. Each of the 5000 points in the figure represents an experiment over 400 queries for a particular operational setting. Not surprisingly, since the TZ method has few parameters and does not model term covariance, it is able to achieve a restricted subset of potential risk-reward tradeoffs. Similarly, for this corpus the Jaccard-based CT expansion method has limited losses, but no real gains. However, the unified method with heat-kernel translation exhibits a broad regime of potential tradeoffs, including a range of gains superior to either baseline algorithm.

4.5 Analysis of additional latent variable conditions

We provide a brief analysis of the example latent variable optimization program described in Sec. 2.5. We evaluated the effect of adding a translation kernel to the objective, and the effect of the linear diversity constraints.

For this summary evaluation, we present results for wt10g

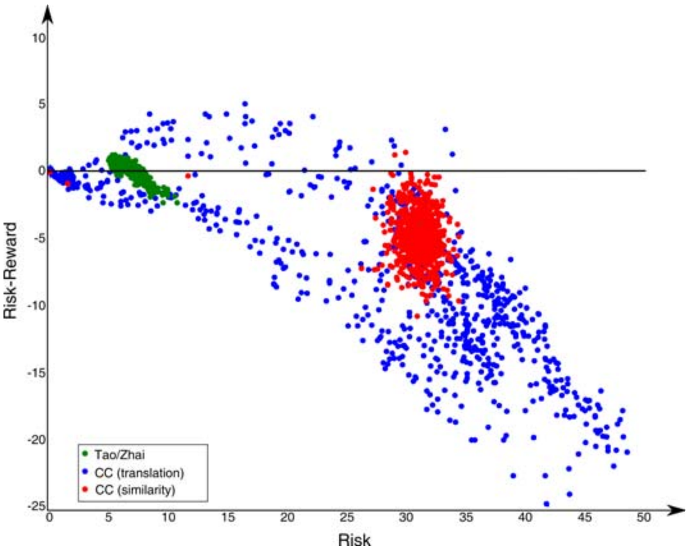


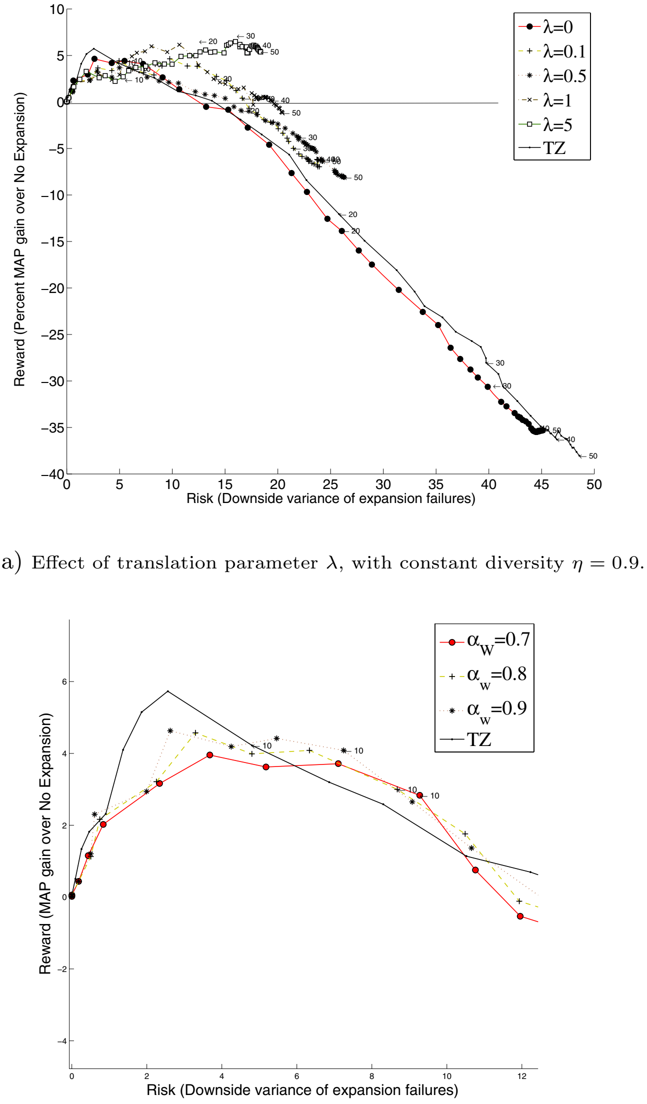
Figure 8: Massive parameter space exploration of the risk-reward tradeoffs achievable in the parameter space of three different query expansion methods: TZ (dense green, left), TZ+CT hybrid with Jaccard translation model (dense red, right) and TZ+CT hybrid with heat-kernel translation model (large scattered blue).

(TREC topics 451-550), using the same setup described in Section 4. We limited the maximum number of iterations for both TZ and Constrained EM to 50, since both algorithms have generally converged by then. We used the 50 top-ranked documents and 20 expansion terms.

Again, we use risk-reward curves to show an algorithm's achievable tradeoff between average precision gain and the loss due to expansion failures. One important difference is that our risk-reward curves are generated as a function of the *number of iterations* of each algorithm instead of functions of a feedback interpolation parameter α.

We set the dilation factor σ^2 and time parameter t of the translation kernel to 0.75 and 5 respectively. In practice, because the initial z_{dw} are very close to zero, for numerical reasons we first run a small number L of iterations (in these experiments $L = 3$) of the standard EM algorithm before switching to the Constrained EM version. Because the diversity constraint is a hard constraint, the convex program is occasionally infeasible at some iteration[4]. When this happens, we simply terminate the search and use the last known good solution.

Figure 9 shows two effects. In a), increasing the amount of translation from $\lambda = 0$ to $\lambda = 5$ gives a dramatic improvement in the risk-reward tradeoff: with the translation kernel on this collection, the algorithm never hurts query performance on average, giving its maximum MAP at convergence. The baseline TZ algorithm, on the other hand, deteriorates quickly after about 20 iterations. In b), making the diversity constraint more strict by decreasing the allowable η percentage available to the top-ranked words acts to flatten the risk-reward curve slightly. Future analysis work includes sensitivity of the results to parameter changes, and measuring interaction effects between diversity and translation or other components. The expansions found by the TZ and constrained EM are quite different, so improved performance is not due to simply slower convergence toward a similar solution.



(a) Effect of translation parameter λ, with constant diversity $\eta = 0.9$.

(b) Effect of diversity parameter η, with no translation ($\lambda = 0$)

Figure 9: Risk-reward tradeoff curves show the effect of increasing the word translation factor λ with fixed diversity constraint $\eta_W = 0.90$, compared to Tao-Zhai baseline (TZ). Curves are a function of iteration, with each iteration as a dot and every 10th iteration numbered. Because the initial query is the starting point and the y-axis shows relative MAP gain, curves will start at the origin and trace out a risk-reward tradeoff with each iteration. Tradeoff curves that are *higher and to the left* are better.

[4] Problems such as infeasibility and failure to converge are detected automatically by the solver and reported with a status code.

5 Discussion and Future Directions

The superior performance of the CCCP algorithm compared to its EM counterpart is interesting considering that both methods are based on the same generative model and are designed to compute a Maximum A Posteriori likelihood solution. Unlike most learning scenarios, however, it appears that for the problem of query expansion, the *nature of the path* on the way to the likelihood objective is much more important than the goal itself. The slow, coordinate-wise ascent approach of EM, and more controlled search path of CCCP approaches, turn into an advantage in such cases. Our CCCP method generalizes the EM approach so that we have finer control over the nature of the steps. It also uses a tighter lower bound than the closed-form EM version that includes an extra entropy term to be maximized. While a bit more involved to optimize, this extra regularization penalty over the distribution of Z appears to help stabilize the solution. We did not do extensive parameter tuning in our model, so further gains may be possible.

While most computational effort for modeling query intent is appropriately spent training large-scale models offline, Web search engines must operate an increasingly complex decision environment in which some evidence, such as user interaction feedback, is only visible at query time. Thus, we forsee that an on-line learning component that solves query-specific optimization problems in real time will be a powerful complement to off-line training. Applications include time- or context-sensitive noise reduction, real-time feature selection on predictor inputs, and 'course correction' via posterior constraints on predictor outputs. Because of their generality and simplicity we believe that CCCP-type frameworks are a good starting point for future exploration of useful constraints and objectives in such problems. In general, we believe that the exploration of effective optimization frameworks opens up a new area of research in information retrieval in which tradeoffs and principled decision-making under uncertainty can be greatly improved.

6 Conclusions

We have made both algorithmic and empirical contributions to the problem of searching for an optimal query model that is both effective and reliable. We used the Convex-Concave procedure as a way to reveal more of the implicit structure, objectives and constraints of an existing feedback algorithm, to unify two complementary approaches, and to generate new algorithms in a principled way. On the empirical side, we performed an iterative performance analysis as well as a risk-reward parameter-space analysis on a high-performance computing cluster to gain new insights into the space of computational tradeoffs achievable with different types of algorithms. A general trend in software systems is that simpler or more powerful algorithms are eventually preferred over methods designed for efficiency in special cases. We believe the advantages of effective optimization frameworks for use in information retrieval systems will soon outweigh their moderate computational costs.

Acknowledgements

We thank Guy Lebanon for his valuable feedback, Tao Tao for helpful discussions of the TZ model, and several anonymous reviewers for their comments.

7 References

[1] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

[2] K. Collins-Thompson. Accounting for stability of retrieval algorithms using risk-reward curves. In *Proceedings of SIGIR 2009 Workshop on the Future of Evaluation in Information Retrieval*, pages 27–28.

[3] K. Collins-Thompson. Reducing the risk of query expansion via robust constrained optimization. In *Proceedings of CIKM 2009*, pages 837–846.

[4] K. Collins-Thompson. Estimating robust query models using convex optimization. In *Advances in Neural Information Processing Systems 21 (NIPS)*, 2008.

[5] K. Collins-Thompson. *Robust model estimation methods for information retrieval*. PhD thesis, Carnegie Mellon Univ., 2008.

[6] K. Collins-Thompson and J. Callan. Query expansion using random walk models. In *CIKM 2005*, pages 704–711.

[7] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley & Sons, second edition, 2005.

[8] F. Diaz. *Autocorrelation and Regularization of Query-Based Information Retrieval Scores*. PhD thesis, University of Massachusetts, Amherst, 2007.

[9] J. Dillon, Y. Mao, G. Lebanon, and J. Zhang. Statistical translation, heat kernels, and expected distances. In *Proc. of UAI 2007*, 2007.

[10] M. Figueirdo, J. Bioucas-Dias, and R. Nowak. Majorization-minimization algorithms for wavelet-based image restoration. *IEEE Transactions on Image Processing*, 16(12):2980–2991, 2007.

[11] J. Graça, K. Ganchev, and B. Taskar. Expectation maximization and posterior constraints. In *NIPS*, 2007.

[12] D. Harman and C. Buckley. The NRRC Reliable Information Access (RIA) workshop. In *Proceedings of SIGIR 2004*, pages 528–529, New York, USA, 2004.

[13] V. Lavrenko. *A Generative Theory of Relevance*. PhD thesis, University of Massachusetts, Amherst, 2004.

[14] Lemur. Lemur toolkit for language modeling & retrieval. 2002. http://www.lemurproject.org.

[15] Q. Mei, D. Zhang, and C. Zhai. A general optimization framework for smoothing language models on graph structures. In *Proceedings of SIGIR 2008*, pages 611–618, 2008.

[16] R. Neal and G. E. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. In *Learning in Graphical Models*, 1998.

[17] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *Proceedings of SIGIR 1998*, pages 275–281, 1998.

[18] B. Sriperumbudur and G. Lanckriet. On the convergence of the concave-convex procedure. In *NIPS 22*. Cambridge, MA: MIT Press, 2009.

[19] T. Strohman, D. Metzler, H. Turtle, and W. B. Croft. Indri: A language model-based search engine for complex queries. In *Proc. Int. Conf. on Intel. Analysis*, 2004.

[20] T. Tao and C. Zhai. Regularized estimation of mixture models for robust pseudo-relevance feedback. In *SIGIR 2006*, pages 162–169.

[21] R. Udupa, A. Bhole, and P. Bhattacharya. A term is known by the company it keeps: On selecting a good expansion set in pseudo-relevance feedback. In *Proceedings of ICTIR 2009*, Advances in IR Theory. Springer, 2009.

[22] Z. Xu and R. Akella. A new probabilistic retrieval model based on the Dirichlet compound multinomial distribution. In *SIGIR 2008*, pages 427–434, 2008.

[23] C.-N. J. Yu and T. Joachims. Learning structural SVMs with latent variables. In *International Conference on Machine Learning (ICML)*, 2009.

[24] A. L. Yuille and A. Rangarajan. The Concave-Convex Procedure (CCCP). In *NIPS 2001*, pages 1033–1040, 2001.