Learning to Re-Rank: Query-Dependent Image Re-Ranking Using Click Data

Vidit Jain
Yahoo! Labs
viditj@yahoo-inc.com

Manik Varma
Microsoft Research India
manik@microsoft.com

ABSTRACT

Our objective is to improve the performance of keyword based image search engines by re-ranking their original results. To this end, we address three limitations of existing search engines in this paper. First, there is no straightforward, fully automated way of going from textual queries to visual features. Image search engines therefore primarily rely on static and textual features for ranking. Visual features are mainly used for secondary tasks such as finding similar images. Second, image rankers are trained on query-image pairs labeled with relevance judgments determined by human experts. Such labels are well known to be noisy due to various factors including ambiguous queries, unknown user intent and subjectivity in human judgments. This leads to learning a sub-optimal ranker. Finally, a static ranker is typically built to handle disparate user queries. The ranker is therefore unable to adapt its parameters to suit the query at hand which again leads to sub-optimal results. We demonstrate that all of these problems can be mitigated by employing a re-ranking algorithm that leverages aggregate user click data.

We hypothesize that images clicked in response to a query are mostly relevant to the query. We therefore re-rank the original search results so as to promote images that are likely to be clicked to the top of the ranked list. Our re-ranking algorithm employs Gaussian Process regression to predict the normalized click count for each image, and combines it with the original ranking score. Our approach is shown to significantly boost the performance of the Bing image search engine on a wide range of tail queries.

Categories and Subject Descriptors

H.3.3 [**Information Retrieval**]: Search

General Terms

Algorithms, Experimentation, Theory

Keywords

Image Search, Image Re-ranking, Click Data

Copyright is held by the International World Wide Web Conference Committee (IW3C2). Distribution of these papers is limited to classroom use, and personal use by others.
WWW 2011, March 28–April 1, 2011, Hyderabad, India.
ACM 978-1-4503-0632-4/11/03.

1. INTRODUCTION

Keyword-based image search is not only a problem of significant commercial importance but it also raises fundamental research questions at the intersection of computer vision, natural language processing, machine learning, and information retrieval. Our objective is to improve the performance of tail queries in image search engines by leveraging click data aggregated across users and sessions.

We address three limitations of existing image search engines in this paper. First, since the query is provided as text rather than an image, it is difficult to automatically bring visual features into play and build on advances in computer vision research. As a result, search engines are forced to rely on static and textual features extracted from the image's parent web page and surrounding text which might not describe salient visual information. Second, image rankers are trained on data with label noise which leads to learning rankers with poor generalization performance. Third, a static ranker is learnt for all query classes within a vertical. The learnt ranker therefore does not adapt its parameters to cope with the very diverse set of user queries.

We propose to mitigate these problems by re-ranking the original search engine results based on user click data. For a given query, we identify the images that have been clicked on previously in response to that query. A Gaussian Process (GP) regressor is then trained on these images to predict their normalized click counts. Next, this trained regressor is used to predict the normalized click counts for the top ranked thousand images. The images are finally re-ranked by sorting on the basis of a linear combination of the predicted click counts and the original ranking scores.

Our re-ranking method tackles all of the three issues highlighted above as follows. First, the GP regressor is trained on not just textual features but also visual features extracted directly from the set of previously clicked images. As a result, images that are visually similar to the clicked images in terms of measured shape, color, and texture properties are automatically ranked highly. Note that our approach does not require any user intervention, restrictions on the query semantics, or any assumptions about the user intent or behavior.

Second, by learning from user click data, we mitigate the label noise problem and difficulties associated with understanding the user's intent. In more detail, rankers generally obtain training data by having human experts label the relevance of query-image pairs on an ordinal scale. However, keywords form an impoverished means of communication between a user and a search engine. It can become extremely

difficult for someone else to fathom the user's intent based on the query keywords alone. As such, even so called "experts" often find it hard to judge the relevance of an image to a query, which results in training sets with label noise. For example, given the query "night train", experts tend to label images of trains at night as being highly relevant and images of motorcycles as non-relevant. However, empirical evidence suggests that most users wish to retrieve images of the Night Train model of the Harley-Davidson motorcycle. Similarly, in response to the query "fracture", experts tend to rate both the images of broken bones as well as the movie Fracture as being highly relevant. Again, empirical evidence suggests that very few users wish to see images from the movie (they would've searched for "fracture movie" or "movie fracture" had they wished to do so).

Expert labels might therefore be erroneous. In fact, they might not even be consistent, with different experts assigning different levels of relevance to the same query-image pair. Such factors bias the training set and the ranker is learnt to be sub-optimal. While there is research aimed at tackling this problem directly [2], our click based re-ranker provides a useful alternative. We hypothesize that, for a given query, most of the previously clicked images are highly relevant and can be leveraged to mitigate the inaccuracies of the baseline ranker.

Third, since the GP regressor is trained afresh on each incoming query, it is free to tailor its parameters to suit the query at hand. For example, images named `TajMahal.jpg` are extremely likely to be of the Taj Mahal. The query-image file-name match feature can therefore be important for landmark queries. On the other hand, the same feature is uninformative for city queries – while images of Delhi's tourist attractions are sometimes named `delhi.jpg` so are random photographs that happen to be the single ones taken during a day trip to the city. Our GP regressor learns this directly from the click training data and weights this feature differently in these two situations. A single static ranker would be inadequate.

Our approach rests on the key assumption that, for a given query, clicked images are highly correlated with relevant images. This assumption is not valid for documents in general [20]. Most web search engines display only a two-line snippet for each document in response to a query. This snippet might not contain enough information for the user to determine if the document answers the query. In these cases, relevance can only be determined by clicking on, and going through, the document. Thus clicked documents might not be relevant to a query. However, in the case of images, most search engines display results as thumbnails. The user can see the entire image before clicking on it. As such, barring distracting images and intent changes, users predominantly tend to click on images that are relevant to their query. Figure 1b highlights this phenomenon by showing some of the most clicked images in response to a few queries sampled from our query logs. Thus our proposed solution is specific to image search and may not work well for documents. The case of videos needs to be studied further. Search engines that display results as single frames with some text will encounter the same issues as documents. Whereas for search engines that display a 30-second clip on a mouse-over, the correlation between clicks and relevance will depend on how representative the clip is of the full video.

To the best of our knowledge, this paper represents the first effort in leveraging user click data for query-dependent image re-ranking (see Section 2 for comparison to previous work). Sophisticated user-click models, based on eye tracking and other studies, have been used for re-ranking in document search to overcome position bias and account for user behavior in a session. Corresponding models do not yet exist for image search and porting them from document search is non-trivial. One of the major challenges that we therefore face is getting image re-ranking to work using only raw click data. The main technical difficulties arise from learning with only positively labeled, very sparse data in high dimensional spaces. Our combined textual, visual, and static feature vector can be more than three thousand dimensional while we would like our method to work for queries with as few as ten clicked images. Preventing over-fitting and ensuring good generalization performance becomes key and we achieve this via a combination of dimensionality reduction and regression techniques. In particular, we demonstrate that using PCA and Gaussian Process regression we can significantly improve the performance of a baseline search engine, such as Bing image search, from an nDCG of 0.6854 to 0.7692 for a diverse set of queries.

The rest of this paper is organized as follows. We discuss related work in Section 2 and focus on previous efforts on image re-ranking as well as the use of click data. Section 3 presents the data set on which we evaluate performance. Section 4 then explores a baseline strategy, called Click Boosting, for re-ranking using click data. Click Boosting promotes all the clicked images to the top of the ranked list and maintains the original ranking everywhere else. We show that while Click Boosting can improve a search measure such as nDCG it has many undesirable consequences. These can be overcome by learning a re-ranker that promotes not just previously clicked images but also images that are likely to be clicked. Details of our proposed GP regression method including features, click count prediction, and re-ranking are presented in Section 5. We conclude in Section 6.

2. RELATED WORK

In this Section, we survey related work on image re-ranking and the use of click data. We do not survey the learning to rank literature [6] as our technique can ride piggy back on any baseline ranker and does not depend on the specific ranking algorithm details. We also do not survey content based image retrieval techniques [28], where an image is given as query rather than keywords, as they are not directly relevant to our work. Finally, we do not survey relevance feedback mechanisms as our method requires no user intervention once the query has been issued. This is motivated by empirical evidence suggesting that only an extremely small number of image search users are willing to provide any form of feedback, including marking relevant or non-relevant images, to improve their search results.

Image search and re-ranking.

One of the key challenges in keyword based image search is converting the textual query into a form amenable for visual search. Image annotation and labeling methods reverse this problem and tag the images in the database with keywords that can then be used for retrieval (see [13, 30, 32] for references and discussions on linking keywords to images). In a closed world, where all possible keywords and queries

are known, one can hope to gather training data for each query and learn query-specific classifiers and rankers.

Techniques that feed the keywords to a text based search engine and re-rank the returned images on the basis of visual (and textual) similarity are more relevant to our work. Pseudo-relevance feedback assumes that the top ranked results are relevant to the query and can be used as positive data for training a re-ranker. Negative training data can be obtained from the bottom of the list [33]. Unfortunately, these assumptions do not hold for general image queries. Current image rankers are not yet good enough to return only relevant images at the top and non-relevant ones at the bottom.

To address this issue, Fergus *et al.* [12] use RANSAC to sample a training subset with a high percentage of relevant images. A generative constellation model is learnt for the query category while a background model is learnt from the query "things." Images are re-ranked based on their likelihood ratio. Observing that discriminative learning can lead to superior results, Schroff *et al.* [26] first learn a query independent text based re-ranker. The top ranked results from the text based re-ranking are then selected as positive training examples. Negative training examples are picked randomly from other queries. A binary SVM classifier is then used to re-rank the results on the basis of visual features. This classifier is found to be robust to label noise in the positive training set as long as the non-relevant images are not visually consistent. Better training data can be obtained from online knowledge resources if the set of queries is restricted. For instance, Wang *et al.* [30] learn a generative text model from the query's Wikipedia page and a discriminative image model from the Caltech and Flickr data sets. Search results are then re-ranked on the basis of these learnt probability models. Some user interaction is required to disambiguate the query.

Clustering based approaches [3, 34] typically assume that the set of relevant images form the largest cluster amongst the top ranked results. Images could then be re-ranked by their distance from the largest cluster. A similar strategy was employed by Berg *et al.* [4]. Instead of simply selecting the largest cluster, they require the user to mark each cluster as relevant or non-relevant. Individual images are labeled optionally to overcome clustering inaccuracies. A re-ranker is then learnt from this positive and negative data.

Most of the methods reviewed so far concentrate on queries which are nouns. It is unclear whether their assumptions will be met and whether they will extend to more complex real-world queries such as "child drinking water" and "300 workout", or queries for which training data can not be obtained from Wikipedia and Flickr. More importantly, none of the methods address the issue of determining user intent and most methods require some form of user intervention.

Click data.

As a surrogate for relevance judgments, implicit feedback from the users of a search engine has been investigated by several researchers. Click-through data, as implicit feedback, has been extensively analyzed in the context of web document search [5, 11, 14]. Joachims *et al.* [17] did an eye tracking experiment to observe the relationship between clicked links and the relevance of the target pages. Radlinski *et al.* [20] concluded that the click-through data is not reliable for obtaining absolute relevance judgments, and is also affected by the retrieval quality of the underlying system. Later experiments on re-ranking web search results [27] and determining correlation between click-through data and the quality of underlying search systems [24] presented more evidence against the utility of click-through data for obtaining absolute relevance judgments. However, as reported by Joachims *et al.* [17], the click-through data can be used to obtain relative relevance judgments i.e., if document A is *preferred* over document B for a given query. They also developed a set of rules and strategies to generate these preference hypotheses, which could be used to estimate the relevance labels [2, 22]. Another approach for interpreting the click-through data is based on the prior research on understanding user goals in web search [9, 23], where separate models for click data are used for different classes of queries – *navigational* and *informational* queries [15].

Click-through data for image search, on the other hand, has been found to be very reliable [8, 29]. Compared to 39% relevant documents among the targets of the clicked links for web search [1], 84% of the clicked images were found to be relevant. Nevertheless, it would appear that the only work leveraging click data for query-dependent image search is [8]. The method builds a query-image click graph and performs backward random walks to determine a probability distribution over images conditioned on the given query which can be used for ranking (not re-ranking). However, the method ranks using nothing but click data and does not look at the image content or text content to determine relevance to the query. As such, their ranker is very different in feel and spirit to our proposed re-ranker. For other ways of building a similarity graph, though not using clicks, and leveraging random walks please see [16, 31].

3. DATA SET, CLICKS, HYPOTHESIS TESTING, & EVALUATION

We sampled a set of 349 queries from the Bing query logs. Since our focus is on low-frequency queries, particularly tail queries, we removed the popular queries such as celebrities, music bands, *etc.* For each of the remaining 236 queries, we retrieved the top thousand images from the Bing image search engine.

For each of the thousand retrieved images for a given query, we also mined the Bing click logs to determine how often the image had been previously clicked on in response to that query. Note that the clicks are being aggregated over users and sessions. This is done for primarily two reasons. First, aggregating clicks across users helps reveal the various interpretations of a query and determine their relative importance. Second, since we are primarily interested in tail queries, the click signal is bolstered by aggregating over as many users as possible. We found that, typically, the number of clicked images for a query varied between zero and a couple of hundred. We also found that some of the clicked images had an extremely high click count and hence we normalized the clicks by taking their logarithm.

We further reject the queries for which the number of clicked images is less than ten. As a result, our final data set is composed of 193,000 images and 193 distinct queries. Acquiring reliable relevance judgments for this large set of image-query pairs is a formidable task. To circumvent this issue, we chose a small set of 19 queries to form a development set, while keeping all of the 193 queries aside for

the final evaluation. The queries in the development set are chosen as representatives from various query classes – animals (giraffe, gnats), maps (pacific ocean, red sea), places (Bern), polysemy (fracture), specific entity (rackets, binder), and specific concept (child drinking water), *etc.* We annotated all of the corresponding thousand retrieved images for each of these queries. For the remaining queries, we only annotated the top 20 images in the ranked lists obtained by different approaches during the final evaluation.

Every query-image pair was annotated very carefully on a four point ordinal scale: 3 – Highly Relevant; 2 – Relevant; 1 – Non-relevant; and 0 – unavailable image (i.e. the image was no longer on the web when we tried downloading it). Note that obtaining just these annotations was very time-consuming. We first had to determine the user intent behind each query by figuring out its meaning(s), checking web documents and Wikipedia pages, and analyzing the set of clicked images. Next, labels were assigned on the basis of not only the image but also the image's parent web page as not all relevance judgments can be made by visual inspection alone.

Note that the number of queries in our development set alone is comparable to the previous image re-ranking methods – between 10 and 20 in [3, 4, 26, 30, 34], 26 in [31], and 45 in [8]. Although Jing *et al.* [16] used about 1000 queries in their experiments, the actual evaluation of the full ranked lists was done only for 210 queries. These queries are related only to products and landmarks, whereas we have no such restriction on the queries used in our experiments. The evaluation set we use (193 queries) is comparable in size and diversity to other data sets that have been used to evaluate image re-ranking algorithms.

To compare the performance of different approaches on the development set, we use tests of statistical significance to mitigate the effects of small development set size. Our null hypothesis is of the form "The performance of algorithm X is indistinguishable from the baseline Bing image search system." We use the T-test to compute the probability (or p-value) with which to accept or reject the null hypothesis. A low p-value suggests that the null hypothesis can be rejected with high probability – i.e. the performance difference between algorithm X and Bing image search is unlikely to be due to chance sampling biases.

We evaluate ranking performance using the standard nDCG search measure. Given the ground truth relevance list $\mathbf{R}$ of a predicted ranking, the Discounted Cumulative Gain at position p is given by $\mathrm{DCG}_p(\mathbf{R}) = \sum_{i=1}^{p}(2^{R_i} - 1)/\log(i+1)$. This measure is sensitive to the rankings in the first p positions with results at the top being given more weight. The query normalized DCG at position p can now be defined as $\mathrm{nDCG}_p(\mathbf{R}) = \mathrm{DCG}_p(\mathbf{R})/\mathrm{DCG}_p(\mathbf{I})$ where $\mathbf{I}$ is the relevance labeling of the *ideal* ranking. We compute mean nDCG at $p = 20$ thereby concentrating on the first page of results since users seldom look further.

4. CLICK BOOSTING

In this Section, we describe the Click Boosting technique which is a straightforward way of re-ranking search results based on click data. This technique promotes all of the clicked images, sorted in descending order according to the number of clicks, to the top. The original ranking is used to break ties as well as to rank all images that have not been clicked. Figure 1 shows the top 5 ranked images for Bing image search as well as Click Boosting for three representative queries.

(a) Bing image search

(b) Click Boosting

Figure 1: ***The top five ranked images for (a) Bing image search and (b) Click Boosting for the queries (from top to bottom): "child drinking water", "Spring Break 2007", and "goggle".***

As can be seen in Figure 1, Bing's performance is variable for different query classes. The ranker is based purely on static and textual features derived from the image's parent web page. The results can be fairly good when the surrounding text matches the image's content as for "child drinking water." However, when there is a mismatch, the results can be quite some way off. For instance, the retrieved results for the query "goggle" have images related to the Japanese TV series "goggle five", and a competition for bartenders named "foggygoggle". Although these terms have textual similarities, they are visually and semantically different from goggles.

For Click Boosting on most of the queries, we observed that the number of times a retrieved result is clicked is highly correlated with the relevance of that image to the query. However, the click data becomes unreliable for distracting images that stand out from the rest of the results (e.g., bottom row, third column for the query "goggle"), perhaps raising interest in the user irrespective of the original information need. We also observed that the click signal was not reliable for images with very few clicks. This is reflected in the three queries in Figure 1. As can be seen, while the results for "child drinking water" remain unaffected, the results for "Spring Break 2007" get noticeably better due to the good click signal while the results for "goggle" get even worse due to image that are distracting or have very few clicks. In terms of quantitative evaluation, Click Boosting improves mean nDCG@20 from the Bing baseline of 0.6854 to 0.7377 on the development set. The T-test suggests that

the null hypothesis, i.e. that Click Boosting's performance is the same as Bing image search's, can be rejected at $p = 0.02$. In other words, there is only a 2% probability that Click Boosting's results are better than Bing's due to chance variations in our data set.

Click Boosting might therefore appear rather attractive at first blush. It is simple to implement, computationally inexpensive, compatible with most search architectures and leads to significant gains in nDCG. However, it has undesirable long term consequences. First, there is the danger of getting into a self referring loop. Images with a large number of clicks will be promoted to the top and get even more clicks as a result. Such images will get entrenched at the top and will be difficult to dislodge. Second, non-relevant distracting images, which might have received only a single, or very few, clicks will also get promoted to the top. Third, relevant unclicked images will forever languish at the bottom and will not be displayed. Thus, Click Boosting is not an acceptable solution. As we demonstrate in the next Section, not only does our proposed re-ranking algorithm overcome these issues but it also leads to further gains in nDCG.

5. GAUSSIAN PROCESS RE-RANKING USING CLICK DATA

The nDCG gains achieved by Click Boosting suggests that click data is useful for improving the baseline search engine results. In this Section, we develop re-ranking techniques which aim to promote not only previously clicked images to the top of the ranked list but also images that are likely to be clicked. To put in context, we are working with clicks aggregated over users and sessions and our objective is to improve performance for tail queries with at least 10 clicked images. Our objective is not to model an individual user's click behavior in order to improve her particular search experience, or to clean the click signal or to infer labels for training the initial ranker.

Our main assumption is that clicked images are highly correlated with relevant images. This is not to say that images with very few clicks are necessarily relevant or that images with no clicks are necessarily non-relevant. In fact, our very objective is to suppress non-relevant distracting images which have very few clicks and promote relevant unclicked images to the the top.

The key technical challenge in doing so arises from the fact that the click data is very sparse. For many of the queries we considered, a handful of relevant images had received a large number of clicks but the total number of clicked images ranged typically from ten to a hundred. Learning high dimensional ranking models becomes very challenging in such scenarios and none of the discriminative methods that we tried worked well. What finally gave good results was a combination of dimensionality reduction and regression techniques.

Our algorithm works as follows. Once a query has been issued, we retrieve the top thousand results from the baseline search engine and extract features. We then identify the set of clicked images and perform dimensionality reduction on all the feature vectors. A Gaussian Process regressor is trained on the set of clicked images and is then used to predict the normalized click counts (pseudo-clicks) for all images. Re-ranking is then carried out on the basis of the predicted pseudo-clicks and the original ranking score.

Our proposed solution is supervised but completely automatic. No human intervention is required as the training labels are inferred from the click data. Human relevance judgments are required only for evaluation. Since our method does not require a separate training set, we use the mean of nDCG@20 values for all of the 19 queries in the development set in our comparisons. The baseline Bing image search engine obtains a mean nDCG@20 of 0.6854 while Click Boosting achieves 0.7377. We demonstrate that our proposed algorithm boosts results to 0.7693 – an extremely significant gain over both the baseline ranker as well as Click Boosting. Note that the larger set of 193 queries is used only in the final evaluation, which will be discussed on Section 5.5. We now explain the various components of our algorithm in more detail.

5.1 Features

We compute the following three types of features from each retrieved image and its parent web page:

- **Query-independent static features** – Several features, such as a document's PageRank, are computed for the parent document that are designed to capture its importance and reliability.
- **Textual features** – A document is likely to be relevant to a given query if the title of the document or the file-name of the image matches the query. For a given query, many such features are computed from the parent document including the image file-name, caption, URL, and meta-data.
- **Image features** – We compute various types of visual features measuring shape, color, and texture properties. We compute SIFT [18], HOG [10], and LBP [19] features as well as color and texture histograms. The extracted visual feature vector has more than two thousand dimensions.

We refer to the union of the first two sets of features as *textual* features, and the third set of features as *visual* features. When concatenated, they form a feature vector with more than three thousand dimensions. Such a diversity of features is necessary for tackling disparate user queries.

5.2 Dimensionality Reduction

While our combined feature vector is more than three thousand dimensional, we have only between twenty and a hundred positive points available for training. This makes discriminative learning extremely hard and most methods tend to over fit with mean nDCG@20 values well below even the baseline. Furthermore, working in a three thousand dimensional space is computationally quite expensive. We address both issues by seeking a compact, low dimensional representation that facilitates learning from a few samples in a computationally efficient manner.

Note that clicks only indicate that some images might be relevant to the query. We do not *a priori* know the set of non-relevant images. We can not therefore apply discriminative dimensionality reduction techniques out of the box as they require negative training data. One can try and apply techniques from document search to generate the set of non-relevant images but the methods do not transfer well. For example, a common rule in document search

Method	mean nDCG@20	Method	mean nDCG@20	Method	mean nDCG@20
AvgRank	0.6266 (-8.6%)	LR	0.6871 (+0.2%)	T	0.7077 (+3.3%)
CorrScore	0.7209 (+5.2%)	SVR	0.6997 (+2.1%)	V	0.6136 (-10.5%)
CorrClick	0.7409 (+8.1%)	1NN	0.7428 (+8.3%)	TV	0.7347 (+7.2%)
PCA	0.7692 (+12.2%)	GP	0.7692 (+12.2%)	T+V	0.7692 (+12.2%)
	(a)		(b)		(c)

Table 1: ***The performance of various techniques for (a) dimensionality reduction, (b) pseudo-click regression, and (c) re-ranking on the development set.*** **The number in brackets is the relative difference in performance compared to the baseline search engine which has a mean nDCG@20 of 0.6854.**

is that if a lower ranked document is clicked then the preceding un-clicked document is probably not relevant. This rule (pseudo-relevance feedback) does not hold for images displayed on a 2D grid rather than a list and images in the vicinity of a clicked image are just as likely to be relevant as not. Another strategy is to take the bottom ranked results as being non-relevant and form the negative training set from them [33]. Unfortunately, our data contains queries which have as many relevant images on the fiftieth page as on the first page in the baseline ranking. It was also not helpful to pick the negative set from the positive points for other close by queries. Note that even if a large set of negative points could somehow be sampled correctly for accurate learning, it would make online training very computationally expensive.

Instead, we found that much better ranking results could be obtained by unsupervised techniques such as PCA or correlation based feature selection. Table 1(a) lists the performance of four dimensionality reduction and feature selection methods which do not need negative training data. Each method is used to select twenty dimensions or features. The methods are

- AvgRank – re-rank the results according to a single individual feature and calculate the average rank of the clicked images. Choose the top few features with the highest average click rank.
- CorrScore – select features that are highly correlated with the ranking score generated by the baseline search engine.
- CorrClick – select features that are highly correlated with the number of clicks.
- PCA – Principal Component Analysis.

We did not try methods such as kernel PCA or random projections since we are interested in obtaining a compact, low-dimensional representation. The performance of three of the methods is quite good. The best results were obtained using PCA and we use it for all further experiments. None of the methods appeared to be sensitive to the choice of number of dimensions. For each method, performance would first increase sharply and then plateau out. We found that choosing twenty dimensions was a good compromise between choosing a compact representation (for efficient learning) and ranking performance.

5.3 Gaussian Process Regression

Given a compact feature representation amenable for learning, our objective is to estimate an image's normalized click count. Note that predicting the probability that an image will get clicked is an extremely hard problem. Fortunately, we need to solve it with only very coarse precision – sufficient to ensure that the final re-ranking is good. As such, rather than regression, the problem could just as easily be posed as ordinal regression, ranking, classification, *etc.* We implemented all these methods and determined that regression gave the best results. Also note that many methods exist for modeling a user's click behavior and predicting the probability of click for search results and advertisements (see our literature survey as well as [7, 22]). However, none of these methods can be applied directly as we need to predict aggregate user clicks as opposed to individual user behavior. The domains are also very different.

Due to the complex semantics of pseudo-clicks estimates, it is likely that they are a non-linear function of the features used to represent the data. Gaussian Processes provide a Bayesian formulation for non-linear regression, where the prior information about the regression parameters can be easily encoded. This property makes them suitable for our problem formulation.

A Gaussian Process (GP) is a collection of random variables, any finite number of which have a joint Gaussian distribution [21]. The estimated pseudo-clicks y for a given feature vector $\mathbf{x}$ is

$$y(\mathbf{x}) = \mathbf{K}(\mathbf{x}, \mathbf{X}_C)[\mathbf{K}(\mathbf{X}_C, \mathbf{X}_C) + \sigma^2\mathbf{I}]^{-1}\mathbf{y}_C, \qquad (1)$$

where $\mathbf{K}$ is a kernel function returning a kernel matrix, $\mathbf{X}_C$ is a matrix of feature vectors for the clicked images, σ is a noise parameter, $\mathbf{I}$ the identity matrix and $\mathbf{y}_C$ the vector of normalized clicks ($\log(1 + \#\text{clicks})$) of the clicked images. Note that we only use the mean and not the variance estimated by the GP. We use a Gaussian kernel in our experiments. We noticed that the performance of the GP had a weak dependence on σ and set it to 0.3.

Table 1(b) compares the performance of GP regression to Linear Regression (LR), Support Vector Regression (SVR) [25], and 1 Nearest Neighbor (1NN). We used a Gaussian kernel for each of the non-linear methods. As can be seen, the performance of LR and SVR is not very good. On examination of the results, LR seemed to be unable to capture the complex relationship between features and pseudo-clicks. LR and SVR also appeared to be performing poorly for queries that had very few clicked images. GP gave the best results and are used in the following re-ranking experiments.

5.4 Re-Ranking

In the previous section, we described a method to compute pseudo-clicks from image features. In this section, we combine these estimated pseudo-clicks with the original ranking score to obtain the re-ranking score. In particular, we model the re-ranking score s_R as the linear combination

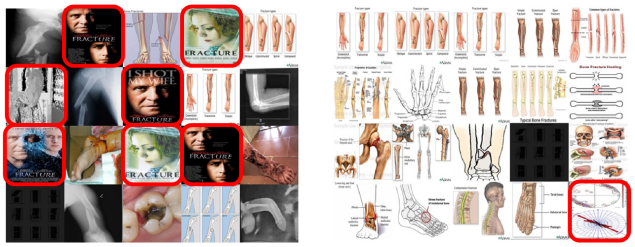
$$s_R(\mathbf{x}) = \alpha\ y(\mathbf{x}_T) + \beta\ y(\mathbf{x}_V) + (1 - \alpha - \beta)\ s_O(\mathbf{x}), \qquad (2)$$

where $\mathbf{x}_T$ and $\mathbf{x}_V$ denote the 20 dimensional projections of the textual and visual features respectively, y is the GP regression function for estimating the pseudo-clicks (see Equation 1), and s_O is the original ranking score. The re-ranking is obtained by sorting s_R in descending order.

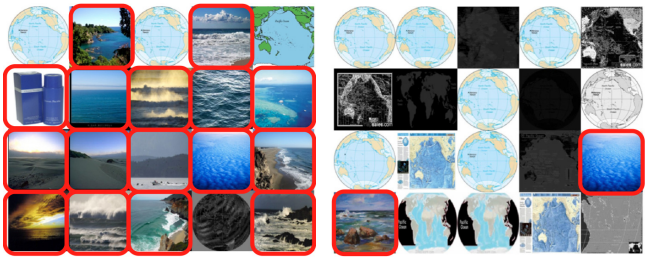
Table 1(c) gives results for various settings of α and β. The Bing image search baseline results have a mean nDCG@20 of 0.6854 and are obtained by setting $\alpha = \beta = 0$. Setting $\beta = 0$ ($\alpha = 0$) results in re-ranking based on textual (visual) features alone corresponding to case T (V) in Table 1(c). As can be seen, re-ranking based on textual features alone improves the mean nDCG@20 to 0.7077. However, the big boost in performance comes from incorporating visual features in addition. The mean nDCG@20 rises sharply to 0.7692 in the T+V. Performing a T-test reveals that the null hypothesis, i.e. Gaussian Process re-ranking using both textual and visual features has the same performance as the Bing image search baseline, can be rejected with a p-value of 0.03. In other words, there is only a 3% probability that our re-ranking results are better than the Bing baseline due to chance variations in the data set. Finally, we also explored the case of using a single joint textual-visual feature representation (case TV in Table 1(c)). In this case, textual and visual feature vectors are concatenated and then projected down to a single 20 dimensional feature vector using PCA. Performance is improved over re-ranking using textual features alone but not as much as projecting textual and visual features separately. This is probably because textual and visual features live in different feature spaces.

Figures 2, 3, and 4 show some qualitative results. Each figure shows the top 20 images, corresponding to the first page of results, for Bing image search (left) and our GP Re-ranking approach (right). Red boxes mark images that are not Highly Relevant to the query. As can be seen, our results are visually significantly better in Figures 2 and 4.

Analyzing the queries in detail, we identified two query categories where performance is improved by re-ranking and three categories where performance is unchanged or becomes worse. It would appear that our method gives the best results when a set of plausible seeming results are actually not desired by users. Figure 2 highlights this situation. As mentioned in the introduction, most users who issue the query "fracture" are actually not interested in images from the movie. This intent is captured in the click data and our re-ranking algorithm prioritizes images of broken bones. Similarly, users searching for "pacific ocean" want to locate it on a map or aerial image. They are seldom interested in picturesque sunsets or images of waves or beaches which could be on any sea or ocean. This is again picked up by our re-ranker which retrieves only a couple of generic ocean images in the top 20. For the quantitative evaluation, we marked images from the movie Fracture and generic ocean images actually of the Pacific, determined from the parent web page, as being Relevant (but not Highly Relevant). Our re-ranker improved the nDCG@20 for "pacific ocean" from 0.5747 to 0.9156 and for "fracture" from 0.7946 to 0.9520. Had we been stricter and marked the above mentioned images as non-relevant then the performance gains would have been even more dramatic. Finally, although it may appear that using visual features may limit diversity, it is important to note that our approach is capable of retaining multiple clusters of images if they have a not insignificant numbers of clicks.

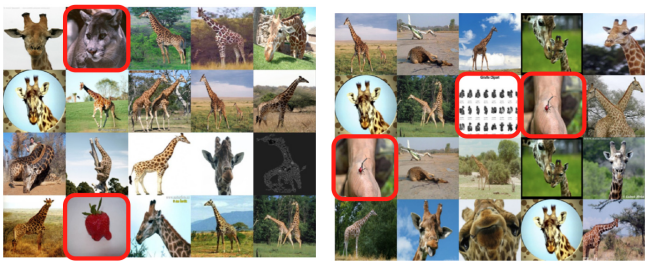


(a) "fracture"



(b) "pacific ocean"

Figure 2: ***Qualitative comparison of Bing image search results (left) and our GP re-ranking approach (right).*** **Re-ranking appears to work well in cases where there is a seemingly plausible query interpretation that is not desired by users. (Red boxes mark images that are not Highly Relevant to the query.)**



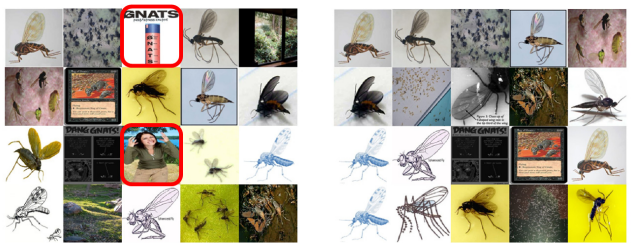
(a) "giraffe"

(b) "rainbow"

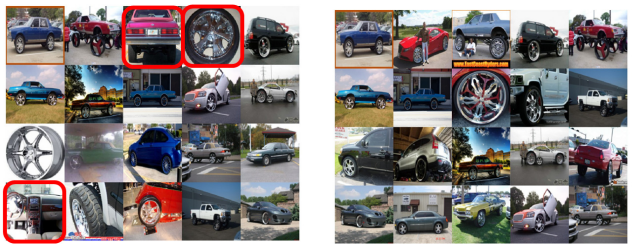
Figure 3: ***Qualitative comparison of Bing image search results (left) and our GP re-ranking approach (right).*** **Re-ranking does not work when there are a lot of relevant, clicked images. (Red boxes mark images that are not Highly Relevant to the query.)**

Figure 4 illustrates another situation where our approach improves performance. In this case, the baseline ranking has a few errors near the top. Our re-ranking algorithm eliminates these by promoting visually similar (as in the case of "gnats") or textually similar (as in "24 inch rims" referring

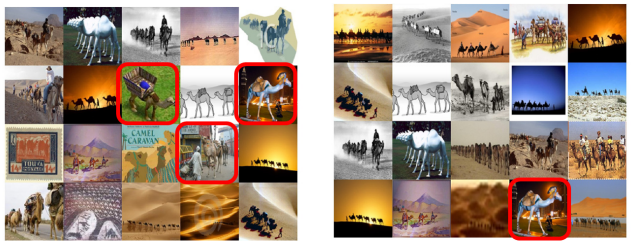
to oversized tires) images to the top ranked clicked images. Both visual and textual features contribute in the case of "camel caravan." Bing's nDCG@20 for the three queries, "gnats", "24 inch rims", and "camel caravan" are 0.7886, 0.8010, and 0.8460 respectively. Our re-ranker boosts these to 0.9193, 0.9589, and 0.9212 respectively. The gains are significant and visually apparent.



(a) "gnats"



(b) "24 inch rims"



(c) "camel caravan"

Figure 4: ***Qualitative comparison of Bing image search results (left) and our GP re-ranking approach (right).*** **Re-ranking can eliminate errors in the original ranking by promoting to the top images that are visually and textually similar to the clicked images. (Red boxes mark images that are not Highly Relevant to the query.)**

Finally, we identified three query classes where re-ranking does not help or makes results worse. Re-ranking does not help much for simple, unambiguous queries where the Bing baseline results are already good. It also does not help much when there are a lot of clicked images which are mostly relevant. In this case, Click Boosting, and even the baseline ranker that includes the number of clicks as a feature, will not be improved by re-ranking. Figure 3 shows results for "giraffe" and "rainbow" both of which have more than 300 clicked images. The nDCG@20 decreased from the Bing baseline of 0.8940 to 0.8910 for "giraffe" and from 0.8320 to 0.7670 for "rainbow" (our worst result). Lastly, re-ranking also made results worse for queries with less than 10 clicked images where most of the images had received only a few clicks and were not strongly correlated with relevance.

5.5 Evaluation on the larger query set

After optimizing the parameters of the re-ranking algorithm on the development set of queries, we evaluated our system on a the set of 193 queries. Figure 5 shows the improvement in the nDCG@20 values with the queries grouped together based on the semantics. In particular, we use ten categories: abstract (broke, education), animals (bear, panda), brand (armani exchange), cartoon (Odie, snoopy), maps (Edmonton map, California state map), places (Chicago, Oslo), polysemy (Turkey, ascot), specific concept (auto accident, optical illusions), specific entity (drums, oboe), and tv (ANTM, starstruck). As shown in Figure 5, our re-ranking algorithm improves the mean nDCG@20 values for all of these query categories.

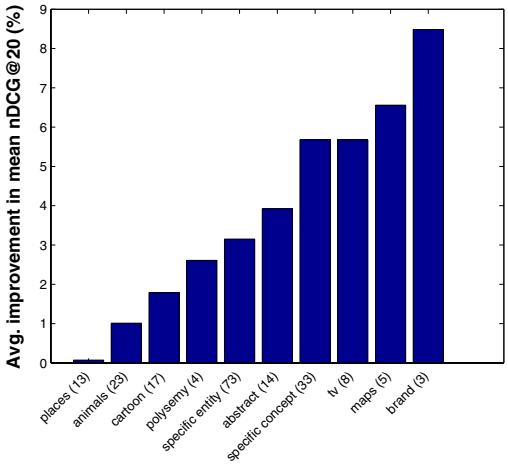


Figure 5: ***Improvement in performance for different query categories.*** **The number of queries in each category is shown in the bracket next to its name.**

Our approach is particularly useful for queries where there is a high correlation between clicks and relevance. This correlation is high when the query concept is easily discernible in the retrieved images, which is likely to be true for queries that refer to specific objects and are visually coherent. "Places" and "animals" are two such query categories. The performance of the base ranker for the queries in these categories were observed to be high, leaving little scope for improvement. For other query categories that satisfy the above properties of visual coherence ("brand", "maps", and "tv") and where the performance of the original ranker is low, we observed big improvements in performance using our approach. Some example queries of these categories are "talk show" (tv), "world map" (maps), and "alcoholic beverages" (specific concept). The other query categories see modest improvements of 2-6% in the mean nDCG@20 values.

For polysemous queries, our approach obtains an improvement of about 3%. Figure 6 shows the re-ranking of the images retrieved for a polysemous query "Turkey". Note that our model favors the images that are visually similar to the clicked images to move up in the ranked list. When multiple interpretations of the given query are manifested as clicked images, multiple clusters of images that are similar to the clicked images can be formed. The images from these clusters compete for the top positions in the re-ranked list. Gaussian process regression can handle arbitrary number of

clusters in the target distribution, and can assign comparable regression scores to images from different clusters. Hence polysemy is preserved in the re-ranking as well.

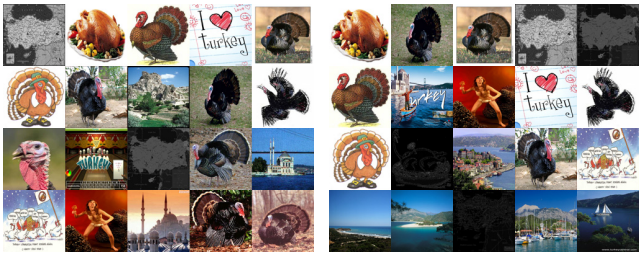


Figure 6: *Polysemy.* The query "Turkey" has multiple interpretations. It may refer to a particular country, a particular bird, or a stuffed turkey prepared for Thanksgiving. The top twenty retrieved images from the base ranker (left) include images for each of these three interpretations. Also, images for each of these meanings were found to be clicked rather uniformly by the users. Our algorithm preserves all of these interpretations in the re-ranked list (right).

For some queries with broad interpretations, our re-ranking algorithm obtained a more visually diverse set of images as compared to the original ranked list, while maintaining the same nDCG values. Figure 7 compares the ranked lists for one such query "Stargate (1994)". This query refers to the movie Stargate that was released in the year 1994. The relevant images for this query include images of scenes from this movie and the poster of this movie. The original ranked list has several near-duplicates of the posters of this movie, whereas the re-ranked list includes the images of the poster and more scenes from the movie. This visual diversification makes the GP ranked list more likely to satisfy the information need of the user than the original ranked list.

Figure 7: *Visual diversity of the ranked list.* Since all of the images in these ranked lists are relevant to the query "Stargate (1994)", the nDCG values are the same for the original ranked list (left) and the re-ranked list (right). The original list contains several near-duplicates (e.g., the second, fourth, and fifth images in the second row). The re-ranked list not only has fewer near-duplicates, but also displays more visually diverse examples of images that are relevant to this query. This visual diversification makes the GP ranked list more likely to satisfy the information need of the user than the original ranked list.

Finally, we grouped the queries based on the number of clicked images. The number of clicked images for a query is positively correlated with the popularity of the query – the tail queries would have very few clicked images in the original ranked list. Figure 8 shows the average improvement in the mean nDCG@20 as a function of the number of clicked images.

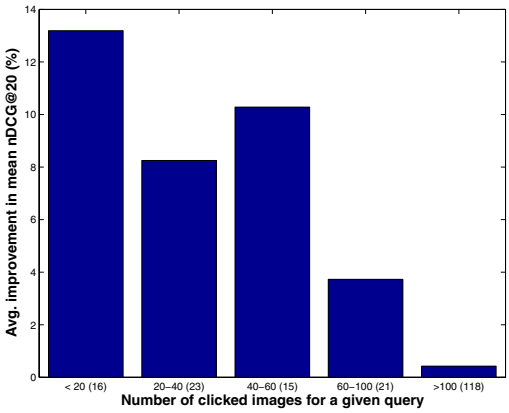


Figure 8: *Improvement in performance as a function of the number of clicked images for a given query.* The number of queries in each bin is shown in the bracket next to the bin label.

Our primary goal was to improve the ranking for tail queries. For queries with less than 20 clicked images (implying these are tail queries), the proposed re-ranking algorithm obtained a huge improvement of about 13% over the Bing image search engine (shown in Figure 8). For queries with more clicked images (subsequent bars), the improvement in performance is expected to be lower in general. However, the monotonicity of this trend can also be affected by the inherent hardness of the queries in these bins as well. For the same grouping of queries, we observed a monotonic decrease in the *absolute* improvement in the nDCG@20 values though.

Gaussian Processes can be computationally expensive in general. This turns out not to be a major concern in our case due to the tiny kernel matrices involved. Given a query with N clicked images, the major computational cost comes from building the $1000 \times N$ matrix encoding the similarity between every clicked image and the top ranked thousand images. This matrix is required by many regression methods apart from GPs. Since N^2 is typically less than a thousand, the cost of building this matrix often dominates the $O(N^3)$ GP cost of inverting the kernel matrix. Thus, even our completely unoptimized Matlab code, running on a standard PC, turns out to be reasonably efficient. For instance, our code takes less than twenty milliseconds to re-rank a query with twenty clicked images.

6. CONCLUSIONS

In this paper, we tackled three major limitations of existing image search rankers – not incorporating visual features, learning from training data with label noise and the use of a static ranker for diverse queries. Focusing on tail queries with more than ten clicked images, we proposed an efficient algorithm that re-ranks baseline results using a lin-

ear combination of the estimated pseudo-clicks and the original ranking score. The main technical challenge of predicting pseudo-clicks from only positively labeled, very sparse, high dimensional data was overcome using dimensionality reduction and Gaussian Process regression. We demonstrated that our proposed algorithm gave significantly better results than not just the production level Bing image search engine but also Click Boosting. The performance gains came from query classes which had ambiguous queries or when there were enough relevant clicked images so that promoting similar images to the top could eliminate errors in the original ranking.

This paper represents one of the first efforts at query-dependent image re-ranking using aggregated click data. Our use of click data differs significantly from other approaches in the recent past which have concentrated on modeling user click behavior in a session. Our approach to image re-ranking also improves over previous work by not requiring the query set to be fixed in advance, not requiring user intervention once the query has been issued and not making very strict and rigid modeling assumptions.

7. ACKNOWLEDGMENTS

We are grateful to Deepak Agarwal, James Allan, Michael Bendersky, Mark Bolin, Qifa Ke, and Nina Mishra for helpful discussions and feedback.

8. REFERENCES

[1] E. Agichtein, E. Brill, and S. Dumais. Improving web search ranking by incorporating user behavior information. In *SIGIR*, 2006.

[2] R. Agrawal, A. Halverson, K. Kenthapadi, N. Mishra, and P. Tsaparas. Generating labels from clicks. In *WSDM*, 2009.

[3] N. Ben-Haim, B. Babenko, and S. Belongie. Improving web-based image search via content based clustering. In *SLAM*, 2006.

[4] T. L. Berg and D. A. Forsyth. Animals on the web. In *CVPR*, 2006.

[5] B. Carterette and R. Jones. Evaluating search engines by modeling the relationship between relevance and clicks. In *NIPS*, 2007.

[6] O. Chapelle, Y. Chang, and T.-X. Liu. ICML 2010 workshop on the learning to rank challenge.

[7] O. Chapelle and Y. Zhang. A dynamic Bayesian network click model for web search ranking. In *WWW*, 2009.

[8] N. Craswell and M. Szummer. Random walks on the click graph. In *SIGIR*, 2007.

[9] E. Cutrell and Z. Guan. What are you looking for?: an eye-tracking study of information usage in web search. In *CHI*, 2007.

[10] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[11] G. Dupret and C. Liao. A model to estimate intrinsic document relevance from the clickthrough logs of a web search engine. In *WSDM*, 2010.

[12] R. Fergus, P. Perona, and A. Zisserman. A visual category filter for google images. In *ECCV*, 2004.

[13] R. Fergus, Y. Weiss, and A. Torralba. Semi-supervised learning in gigantic image collections. In *NIPS*, 2009.

[14] S. Fox, K. Karnawat, M. Mydland, S. Dumais, and T. White. Evaluating implicit measures to improve web search. *ACM TOIS*, 23(2):147–168, 2005.

[15] F. Guo, L. Li, and C. Faloutsos. Tailoring click models to user goals. In *WSCD*, 2009.

[16] Y. Jing and S. Baluja. VisualRank: Applying PageRank to large-scale image search. *IEEE PAMI*, 30(11):1877–1890, 2008.

[17] T. Joachims, L. Granka, B. Pan, H. Hembrooke, F. Radlinski, and G. Gay. Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. *ACM TOIS*, 25(2):7, 2007.

[18] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[19] T. Ojala, M. Pietikainen, and T. Maenpaa. Gray scale and rotation invariant texture classification with local binary patterns. In *ECCV*, 2000.

[20] F. Radlinski, M. Kurup, and T. Joachims. How does clickthrough data reflect retrieval quality? In *CIKM*, 2008.

[21] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, December 2005.

[22] M. Richardson, E. Dominowska, and R. Ragno. Predicting clicks: estimating the click-through rate for new ads. In *WWW*, 2007.

[23] D. E. Rose and D. Levinson. Understanding user goals in web search. In *WWW*, 2004.

[24] F. Scholer, M. Shokouhi, B. Billerbeck, and A. Turpin. Using clicks as implicit judgments: Expectations versus observations. In *ECIR*, 2008.

[25] B. Scholkopf and A. Smola. *Learning with Kernels*. MIT Press, 2002.

[26] F. Schroff, A. Criminisi, and A. Zisserman. Harvesting images databases from the web. In *ICCV*, 2007.

[27] M. Shokouhi, F. Scholer, and A. Turpin. Investigating the effectiveness of clickthrough data for document reordering. In *ECIR*, 2008.

[28] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. C. Jain. Content-based image retrieval at the end of the early years. *IEEE PAMI*, 22(12):1349–1380, 2000.

[29] G. Smith and H. Ashman. Evaluating implicit judgments from image search interactions. In *WebSci'09: Society On-line*, 2009.

[30] G. Wang and D. A. Forsyth. Object image retrieval by exploiting online knowledge resources. In *CVPR*, 2008.

[31] L. Wang, L. Yang, and X. Tian. Query aware visual similarity propagation for image search reranking. In *ACM Multimedia*, 2009.

[32] J. Weston, S. Bengio, and N. Usinier. Large scale image annotation: Learning to rank with joint word-image embeddings. In *ICML Workshop on the Learning to Rank Challenge*, 2010.

[33] R. Yang, A. Hauptmann, and R. Jin. Multimedia search with pseudo-relevance feedback. In *CIVR*, 2003.

[34] H. Zitouni, S. G. Sevil, D. Ozkan, and P. Duygulu. Re-ranking of image search results using a graph algorithm. In *ICPR*, 2008.