Context-Aware Ranking in Web Search

Biao Xiang[1]* Daxin Jiang[2] Jian Pei[3] Xiaohui Sun[2] Enhong Chen[1] Hang Li[2]
[1]University of Science and Technology of China [2]Microsoft Research Asia [3]Simon Fraser University
[1]{bxiang, cheneh}@ustc.edu.cn [2]{djiang, xiaos, hangli}@microsoft.com [3]jpei@cs.sfu.ca

ABSTRACT

The context of a search query often provides a search engine meaningful hints for answering the current query better. Previous studies on context-aware search were either focused on the development of context models or limited to a relatively small scale investigation under a controlled laboratory setting. Particularly, about context-aware ranking for Web search, the following two critical problems are largely remained unsolved. First, how can we take advantage of different types of contexts in ranking? Second, how can we integrate context information into a ranking model? In this paper, we tackle the above two essential problems analytically and empirically. We develop different ranking principles for different types of contexts. Moreover, we adopt a learning-to-rank approach and integrate the ranking principles into a state-of-the-art ranking model by encoding the context information as features of the model. We empirically test our approach using a large search log data set obtained from a major commercial search engine. Our evaluation uses both human judgments and implicit user click data. The experimental results clearly show that our context-aware ranking approach improves the ranking of a commercial search engine which ignores context information. Furthermore, our method outperforms a baseline method which considers context information in ranking.

Categories and Subject Descriptors

H.3.m [**Information Storage and Retrieval**]: Search process

General Terms

Algorithms, Experimentation

Keywords

Context-aware ranking, learning-to-rank application

*The work was done when Biao Xiang was an intern at Microsoft Research Asia.

SIGIR'10, July 19–23, 2010, Geneva, Switzerland.

1. INTRODUCTION

In Web search, given a query, a search engine returns the matched documents in a ranked list to meet the user's information need. Ranking models play a central role in search engines. Currently, almost all the existing ranking models consider only the current query and the documents, but do not take into account any context information such as the previous queries in the same session and the answers clicked on or skipped by the user to the previous queries. In other words, almost all the current ranking models are insensitive to context.

Information Retrieval research has well recognized that context information is very helpful in achieving good search results. Context information may provide hints about users' search intent and help to make better matching with documents. For example, if a user raises a query "*jaguar*" after she searches "*BMW*", it is very likely that the user is seeking for information about a Jaguar car rather than a jaguar as an animal. The absence of context information in document ranking models is probably partially due to the difficulty of obtaining context information. Only recently have large amounts of search session data become available, which enable large scale empirical studies on context-aware methods for Web search.

Several recent studies explore context-aware search methods from different angles. Shen *et al.* [15] presented a context-aware ranking method by assuming that context information can better represent search intent. They incorporated the context information to build context-aware language models, which were assumed to give rise to documents not only similar to the current query but also similar to the previous queries and the summaries of the documents clicked on. The study confirmed the effectiveness of the ranking model on TREC data (`http://trec.nist.gov`). However, the evaluation was based on a small data set consisting of only thirty sessions from three subjects under a controlled laboratory setting. It is unclear whether the assumption in the study holds for Web search engines in the real world.

More recently, Cao *et al.* [2, 3, 4] extracted context information in Web search sessions by modeling search sessions as sequences of user queries and clicks. They learned sequential prediction models such Hidden Markov Model and Conditional Random Fields from search log data. Different from our study here, their models were designed for predicting search intents based on context information, but not for ranking. Therefore, the models are more suitable for query suggestion, query categorization, and URL recommendation than search results ranking, as will be further analyzed in Section 2.

In spite of the several existing studies on context-aware

search methods, the following two critical problems about context-aware ranking for Web search are largely remained unsolved. First, how can we take advantage of different types of contexts in ranking? Second, how can we integrate context information into a ranking model? In this paper, we tackle the above two essential problems analytically and empirically and make the following contributions.

We develop four different ranking principles for different types of contexts. Those principles promote or demote documents according to the context of the current query. We evaluate the four principles using real Web search sessions, and confirm the effectiveness of three principles through the significance test on the data. Interestingly, only one of the three effective principles is consistent with the findings obtained by Shen *et al.* [15] on TREC data, indicating that Web search is quite different from search on TREC data.

Moreover, we adopt a learning-to-rank approach and integrate the ranking principles into a state-of-the-art ranking model, RankSVM [7], by encoding the context information as features. We empirically test our approach using a large search log data set obtained from a major commercial search engine. Our evaluation uses both human judgments and implicit user click data. The experimental results clearly show that our context-aware ranking approach improves the ranking of a commercial search engine which ignores context information. Furthermore, our method outperforms a baseline method which considers context information in ranking.

The rest of the paper is organized as follows. We review related work in Section 2. We discuss the types of contexts, propose ranking principles, and evaluate the effectiveness of the principles in Section 3. We incorporate context information into a learning-to-rank model in Section 4. The experimental results are reported in Section 5. The paper is concluded in Section 6.

2. RELATED WORK

Different users may prefer different results for the same query. *Personalized search* (e.g., [5, 12, 16, 17, 18]) aims to provide the most relevant search results to individual users based on their interests. Traditional personalization approaches usually build a profile of interests for each user from her/his search or browsing history.

Context information is useful in identifying users' search needs. *Context-aware search* adapts search results to individual search needs using contexts. While personalized search considers *individual* users' long and/or short histories, context-aware search focuses on short histories of *all* users. Research on context-aware search has been concentrated on modeling contexts. For example, Cao *et al.* [2] mined frequent sequential patterns from search sessions for context-aware query suggestion. Cao *et al.* [4] modeled contexts containing both queries and clicks within sessions by learning a variable length Hidden Markov Model for query suggestion, URL recommendation, and document re-ranking. Cao *et al.* [3] incorporated context information into a Conditional Random Field (CRF) model for better query classification. Those models were mainly designed for inferring and predicting user search intents using context information. Therefore, they are more suitable for tasks such as query suggestions and query categorization than ranking. Although the generative HMM model in [4] can be applied to search results ranking, it learns parameters for individual queries and contexts. Consequently, the learned HMM model in [4] can hardly be generalized to handle new queries and contexts not occurring in the training data. In this paper, we mainly focus on ranking principles and models which can be generalized to handle new queries and contexts.

Shen *et al.* [15] proposed a method for context-aware ranking, which is probably the work most related to this study. They enriched the current query by using context information, and then fitted the enriched query into language models for retrieval. The basic idea is to promote the documents that are more similar to the previous queries and clicked documents within the same session. The authors verified the effectiveness of the method using a small amount of session data created upon TREC data. In reality, user sessions for Web search are more complex. As indicated in previous studies (e.g, [9, 10, 13]), there are multiple possible relations between the current query and the previous queries, such as reformulation, specialization, generalization, and parallel movement. Considering only one situation as in [15] may not be sufficient in complicated cases.

Our work is fundamentally different from the previous studies in the following two aspects. First, different from the previous work on building context models for user intent understanding [2, 3, 4], our study targets at the problem of context-aware ranking in Web search. To the best of our knowledge, we are the first to systematically explore context-aware ranking in real Web search scenarios. Second, compared to the previous work which applies a single ranking strategy to all kinds of contexts [15], our work recognizes different types of contexts, and proposes corresponding principles.

3. RANKING PRINCIPLES

In this section, we propose context-aware ranking principles according to the relations between the current query and the contexts, and evaluate the effectiveness of the principles using real log data extracted from a major commercial search engine.

3.1 Context-Aware Ranking Principles

In general, the context of a query q being asked contains any information that is related to q and available when q is asked. Specifically, in a Web search engine, the context often contains the queries asked before q in the same session as well as the answers (URLs) to those queries that are clicked on or skipped by the user. In the rest of this section, for a query q_t in a session, we constrain the context of q_t to only the query q_{t-1} asked right before q_t in the session and the answers to q_{t-1} clicked on or skipped by the user. We will extend our consideration of context to all the queries and answers preceding q_t in the session in Section 4.

The relations between queries in sessions have been studied in several previous works [9, 10, 13], which agree on five general types. That is, the current query q_t can be unrelated to, reformulating, specializing, generalizing, or generally associated with the preceding query q_{t-1} in the same session. Obviously, for a query unrelated to its context, the context information cannot help. We discuss the other kinds of relations in this subsection.

3.1.1 Reformulation

A user may reformulate her previous query into a new one because the search results for the previous one do not or only partially fulfill her information need. The user's information need does not change in the case of reformulation.

EXAMPLE 1 (REFORMULATION). Table 1 shows two consecutive queries in a session in a real log data set. The user

Query 1: "homes for rent in atlanta"		Query 2: "houses for rent in atlanta"	
×	**Atlanta homes for rent - home rentals - houses for ren...** Rentlist is directory of Atlanta home rentals featuring links to... *http://www.rentlist.net*		**Atlanta homes for rent - home rentals - houses for ren...** Rentlist is directory of Atlanta home rentals featuring links to... *http://www.rentlist.net*
	Homes For Rent, lease in Atlanta suburbs. Can't sell ... Atlanta homes for rent, homes for lease in Gwinnett and north... *http://atlantahomesforrent.com*		**Homes for Rent in Atlanta, GA** Houses, Apartments and Homes for Rent in Atlanta, GA Find ... *http://www.usrentallistings.com/ga/atlanta*
	Rentals.com - Homes for Rent, Apartments, Houses ... Atlanta Home Rentals; Austin Home Rentals; Charlotte Home... *http://www.rentals.com*		**Atlanta Home Rentals, Homes for Rent in Atlanta ...** Atlanta Rentals - Homes for Rent in Atlanta, Apartments, Re... *http://www.rentals.com/Georgia/Atlanta*
×	**Atlanta Home Rentals, Homes for Rent in Atlanta ...** Atlanta Rentals - Homes for Rent in Atlanta, Apartments, Re... *http://www.rentals.com/Georgia/Atlanta*		**Homes For Rent, lease in Atlanta suburbs. Can't sell ...** Atlanta homes for rent, homes for lease in Gwinnett and north... *http://atlantahomesforrent.com*
	Homes for Rent in Atlanta, GA Houses, Apartments and Homes for Rent in Atlanta, GA Find ... *http://www.usrentallistings.com/ga/atlanta*	×	**Atlanta Homes for Rent, Rental Properties, Houses for ...** Search for Homes for Rent in Atlanta, Georgia for free. View li... *www.rentalhouses.com/find/GA/AtlantaArea/ATLANTA*

Table 1: An example of successive queries with reformulation relation.

Query 1: "time life music"		Query 2: "time life Christian CDs"	
×	**Welcome to TimeLife.com | Homepage** TimeLife.com: The best in music & video from a name you can... *http://www.timelife.com*		**Welcome to TimeLife.com | Homepage** Enjoy 138 romantic classics on 9 CDs from top artists like John... *http://www.timelife.com*
	Time-Life - Wikipedia, the free encyclopedia Time-Life is a creator and direct marketer of books, music, vid... *http://en.wikipedia.org/wiki/Time-Life_Music*		**Time Life Music & Video As Seen On TV** Christian ... Time Life Music & Video CD & DVD Collections ... *http://www.asseenontvmusic.com/timelife.html*
	Welcome to TimeLife.com | Music Shop online for exclusive music CDs, music collections, & musi... *http://www.timelife.com/webapp/wcs/stores/servlet/Categor...*		**Welcome to TimeLife.com | Music** Shop online for exclusive music CDs, music collections, & musi... *http://www.timelife.com/webapp/wcs/stores/servlet/Categor...*
	Contemporary Country (Time-Life Music) - Wiki... Contemporary Country was a 22-volume series issued by Time-... *http://en.wikipedia.org/wiki/Contemporary_Country_(Time-...*	×	**Songs ... Time Life 10 CD Collection... Christian Music** CD/Album review of Songs 4 Ever Time Life 10 CD Collection... *http://www.titletrakk.com/album-cd-reviews/songs-4...*
	Time Life Canada | Homepage The most comprehensive country music collection dedicated to... *http://www.timelife.ca*	×	**Christian Band - Newsong - More Life - CD Review of ...** Christian Band - Newsong - More Life CD Review ... Three yea... *http://christianmusic.about.com/cs/cdreviews/fr/aafpr09080...*

Table 2: An example of successive queries with specialization relation.

Query 1: "Free online Tetris"		Query 2: "Tetris game"	
×	**Tetris Friends Online Games - Play Free Games Featuri...** Play free online games featuring Tetris. Play single-player and ... *http://tetrisfriends.com*		**Tetris Friends Online Games - Play Free Games Featuri...** Play free online games featuring Tetris. Play single-player and ... *http://tetrisfriends.com*
×	**Play Free Tetris Game Online** Play this classic, original, Flash Tetris Game online for free. *http://www.gametetris.com*		**Tetris game** Free online game: Make lines with falling blocks! Russia's finest... *http://www.play.vg/games/6-Tetris.html*
	Free Tetris Game Free tetris game - Play free tetris games online, learn about tet... *http://www.tetrislive.com*	×	**Tetris (Game Boy) - Wikipedia, the free encyclopedia** Tetris was a pack-in title included with the Game Boy at the ha... *http://en.wikipedia.org/wiki/Tetris_(handheld_game)*
	4FreeOnlineGame.com - Free Online Tetris Game 4FreeOnlineGame - Free Online Tetris Game ... This is the all ... *http://www.4freeonlinegame.com/Tetris*	×	**Tetris - non-stop puzzle action** Tetris logo, Tetris theme song and Tetriminos are trademarks of... *http://www.tetris.com*
	Tetris - Play Tetris. Free online games @ Adoption Media, LLC 1995 - 2010 This site should not subst... *http://games.adoption.com/free-online-games/Tetris*		**Free Tetris Game** Free tetris game - Play free tetris games online, learn about tetr... *http://www.tetrislive.com*

Table 3: An example of successive queries with generalization relation.

Query 1: "Xbox 360"		Query 2: "FIFA 2010"	
×	**Xbox.com | Home** Xbox.com is your ultimate source for all things Xbox and Xb... *http://www.xbox.com*		**FIFA.com - The Official Website of the FIFA World Cup** The Official Website of the 2010 FIFA World Cup South Africa™ *http://www.fifa.com/worldcup/index.html*
	Xbox 360 - Wikipedia, the free encyclopedia The Xbox 360 is the second video game console produced by ... *http://en.wikipedia.org/wiki/Xbox_360*		**2010 FIFA World Cup - Wikipedia, the free encyclopedia** The template below has been deprecated (see discussion), and ... *http://en.wikipedia.org/wiki/2010_FIFA_World_Cup*
×	**Xbox.com | Xbox 360** Find out more about Xbox 360, the awesome lineup of games ... *http://www.xbox.com/en-US/hardware*		**FIFA.com - Fédération Internationale de Football Associa...** The official site of the international governing body of the sport ... *http://fifa.com*
	Microsoft Xbox Xbox 360 delivers the most powerful console, the next genera... *http://www.microsoft.com/xbox*	×	**FIFA 10 Soccer : FIFA 2010 - EA Sports Games** Improvement in Management Mode, Flick Passes, Ball Physics, ... *http://www.ea.com/games/fifa-soccer*
	Xbox 360 - Gizmodo This No-Name HTPC Remote Has a Keyboard, Can Work W... *http://gizmodo.com/tag/xbox-360*		**FIFA 2010 World Cup in South Africa** A surprise in the 2007 Asian Cup! The Iraqis win it! In spite of ... *http://southafrica2010.wordpress.com*

Table 4: An example of successive queries with general association.

first raised query "*homes for rent in Atlanta*" and clicked on the 1st and 4th search results. The user then issued the second query "*houses for rent in Atlanta*" and clicked on the 5th search result.

The two queries bear similar meaning. Unsurprisingly, 4 out of the top-5 results returned by the search engine for the second query were also among the top-5 results for the first query. Why did the user skip the top-4 search results for the second query but click on the 5th one?

The 1st and 3rd results for the second query were clicked on by the user for the first query. Obviously, a user was unlikely to click again on pages she just browsed.

Moreover, according to some previous user studies [6, 7, 8], a search result is likely to be *viewed* by a user if it is 1) among the top two search results; 2) ranked above the lowest clicked result; or 3) ranked one position below the lowest clicked result. If a search result is *skipped* (i.e., viewed but not clicked on) by a user, it suggests the result may not be interesting to the user. In Example 1, the 2nd and 4th results for the second query were ranked either above or one position below the lowest clicked result for the first query. They were skipped by the user for the first query, and thus can be regarded uninteresting to the user. Therefore, they were unlikely to be clicked on for the second query, either. ■

PRINCIPLE 1 (REFORMULATION). *For consecutive queries $q_{t-1}q_t$ in a session such that q_t reformulates q_{t-1}, if a search result d for q_{t-1} is clicked on or skipped, d as a result for q_t is unlikely to be clicked on and thus should be demoted.* ■

3.1.2 Specialization

When a user issues a specializing query, she likely wants to see results that are more specific about her interests.

EXAMPLE 2 (SPECIALIZATION). Table 2 shows two consecutive queries. The user first asked "*time life music*" and clicked on the homepage of the store. The user further asked "*time life Christian CDs*" and clicked on the 4th and 5th results.

The information need of the second query consists of two parts: information about "time life" and that about "Christian CDs". If we do not consider the context information, both components should be equally important in ranking search results of the second query. However, given the first query, the user likely wanted to see the search results for the second query specifically about the Christian CDs of the music store. This explains why the user skipped the first three results to the second query where the terms "Christian" and "CDs" do not appear in the titles of the search results. ■

PRINCIPLE 2 (SPECIALIZATION). *For consecutive queries $q_{t-1}q_t$ in a session such that q_t specializes q_{t-1}, the user likely prefers the search results specifically focusing on q_t.* ■

The principle is particularly useful in several scenarios. For example, when q_t is rare and q_{t-1} is popular, the answers fully matching q_{t-1} but partially matching q_t may be ranked high for q_t by a search engine. The principle can use the context information to demote the answers matching q_{t-1} given that q_{t-1} was just asked by the user.

One possible way to implement the principle is as follows. Let $q_t \setminus q_{t-1}$ be the set of terms appearing in query q_t but not in query q_{t-1}. If $q_t \setminus q_{t-1} \neq \emptyset$, we should promote the results matching $q_t \setminus q_{t-1}$ in the set of answers to q_t.

3.1.3 Generalization

A user may ask a query more general than the previous one. In such a situation, the user may like to see some information not covered by the first query.

EXAMPLE 3 (GENERALIZATION). Table 3 shows a generalization scenario. A user first asked query "*free online Tetris game*" and clicked on the 1st and 2nd search results. The user then asked query "*Tetris game*" and clicked on the 3rd and 4th results.

The second query "*Tetris game*" carries multiple possible information needs. For example, the user may want to download the game or play it online. Alternatively, the user may be interested in the history or news of the game. The user may also look for the basic game rules or advanced cheats of the game. For such a query with ambiguous search needs, search engines often try to diversify search results. In this example, the top-5 results can be divided into two groups. The 1st, 2nd, and 5th results link to some free online Tetris game sites, while the 3rd and 4th results are about the background information of the Tetris game.

With the context that the previous query was "*free online Tetris game*" and the user clicked on two related sites, we may infer that the user's interest in the second query may likely divert to something about the game but not the game sites. This may explain why the user clicked on the results about the background information of the game. ■

PRINCIPLE 3 (GENERALIZATION). *For consecutive queries $q_{t-1}q_t$ in a session such that q_t generalizes q_{t-1}, the user would likely not prefer the search results specifically focusing on q_{t-1}.* ■

One possible way to implement the principle is as follows. Let $q_{t-1} \setminus q_t$ be the set of terms appearing in q_{t-1} but not in q_t. If $q_{t-1} \setminus q_t \neq \emptyset$, we should demote the results matching $q_{t-1} \setminus q_t$ among the answers to query q_t.

3.1.4 General Association

When a query (especially an ambiguous one) is generally associated with its context, the context may help to narrow down the user's search intent.

EXAMPLE 4 (GENERAL ASSOCIATION). In Table 4, a user first raised query "*Xbox 360"* and clicked on the 1st and 3rd search results. Then, the user raised query "*FIFA 2010*" and clicked on the 4th result.

The second query "*FIFA 2010*" bears multiple intents. It may refer to either the FIFA 2010 World Cup at South Africa or a new game of Xbox 360. Therefore, the second query "*FIFA 2010*" is generally associated with the first query "*Xbox 360*". Without the context, a search engine may retrieve search results for both intents behind query "*FIFA 2010*". However, using the first query "*Xbox 360*" as the context, which indicates that the user was interested in Xbox 360, we may rank the results about the soccer game in Xbox 360 higher than those about the World Cup event. ■

PRINCIPLE 4 (GENERAL ASSOCIATION). *For consecutive queries $q_{t-1}q_t$ in a session such that q_t and q_{t-1} are generally associated, the user likely prefers the search results related to both q_{t-1} and q_t. Such results should be promoted for q_t.* ■

One possible way to implement the principle is the following. First we can choose any topic taxonomy such as the

Pid	# cases	$P(c=1\|h=1)$	$P(c=1\|h=0)$	Δ
1	1,628	0.361	0.217	0.144*
2	1,378	0.401	0.302	0.099*
3	246	0.339	0.315	0.024
4	4,457	0.352	0.296	0.056*

* Passes the significance test at the confidence level of 0.01.

Table 5: The effectiveness of ranking principles in the corresponding types of contexts.

Pid	# cases	$P(c=1\|h=1)$	$P(c=1\|h=0)$	Δ
1	10,186	0.356	0.234	0.122*
2	20,200	0.407	0.316	0.091*
3	1,539	0.358	0.386	-0.028
4	21,052	0.352	0.318	0.034*

* Passes the significance test at the confidence level of 0.01.

Table 6: The effectiveness of ranking principles in all contexts.

Open Directory Project (`http://www.dmoz.org`). Let C_{t-1} and C_t be the sets of topics of q_{t-1} and q_t, respectively, and $C_\cap$ be the set of common topics between C_{t-1} and C_t. If $C_\cap \neq \emptyset$, we should promote a search result u if the set of topics of u shares at least one topic with $C_\cap$.

3.2 Effectiveness of Principles

We use the search log data from a major commercial search engine to evaluate the effectiveness of the principles. We traced each user's query & click stream by the user-id information in the data. All users were completely anonymous, and no action was taken to reveal the users' identities. We segmented each user's stream into sessions by a commonly applied rule [19]: a boundary between two sessions was set if there was no activity by the user for thirty minutes. From the resulted 37,320 user sessions, we extracted successive query pairs within the same sessions, and manually labeled the relations (i.e., reformulation, specialization, generalization, general association, and unrelated) for $10,000$ randomly selected successive query pairs.

3.2.1 Evaluation in Different Types of Contexts

We first evaluate the effectiveness of each principle in its corresponding type of contexts, i.e., when the two successive queries match the relation of the principle. Table 5 shows the number of successive query pairs for each type of contexts, where "Pid" indicates the principle-id. In our evaluation, a search result u is represented by the terms in its title, snippet, and URL. For Principle 4, we use a classifier in [14] and classify all the queries and documents into the 16 topics at the first level of Open Directory Project (`http://www.dmoz.org`). For each query or document, we keep the top three topics returned by the classifier.

According to the previous studies [6, 7, 8], a user views only a subset of search results and chooses to click on or skip them individually. Therefore, in each test case for a principle, we focus on those search results that are likely to be viewed by the user. Specifically, we adopt the methods in [6, 7, 8] and consider a search result is viewed if it is ranked above or one position below the last clicked result. To evaluate a principle, we aggregate the viewed search results for all queries in the test cases and obtain a set U. We call a search result $u \in U$ *satisfies* the principle if u should be promoted (in cases of Principles 2 and 4) or not demoted (in cases of Principles 1 and 3) by the principle; otherwise, u *violates* the principle. Let $U_{h1} \subseteq U$ be the set of search results that satisfy the principle, and $U_{h0} = U \setminus U_{h1}$. Simultaneously, U can also be divided into two subsets U_{c1} and U_{c0}, where $U_{c1} \subseteq U$ consists of the search results that were clicked on by the users, and $U_{c0} = U \setminus U_{c1}$.

The conditional probability for a search result u to be clicked on for q_t given that it satisfies a principle can be estimated as $P(c=1|h=1) = \frac{|U_{c1} \cap U_{h1}|}{|U_{h1}|}$, where random variable c denotes whether a search result u was clicked on for q_t or not, and random variable h denotes whether u satisfies the principle or not. Analogously, the conditional probability for u to be clicked on for q_t given that it violates a principle can be estimated as $P(c=1|h=0) = \frac{|U_{c1} \cap U_{h0}|}{|U_{h0}|}$.

We conduct a t-test on $\Delta = P(c=1|h=1) - P(c=1|h=0)$, the difference between the two conditional probabilities. Intuitively, for each principle, this difference indicates how likely users would choose to click on a search result satisfying instead of violating the principle. One may wonder whether user clicks contain position bias. Since Δ value calculates the difference between two click probabilities, we may expect that the position biases are canceled out. Therefore, if the difference Δ passes the significance test at confidence level 0.01, it confirms the effectiveness of the principle. From Table 5, we can see that Principles 1, 2, and 4 pass the significance test, which supports their effectiveness. However, Principle 3 does not pass the significance test. One reason is that generalization pairs are relatively rare, only 2.46% in the manually labeled data. We can hardly draw reliable conclusions from such a small size of test data.

3.2.2 Evaluation in All Contexts

Given two consecutive queries $q_{t-1}q_t$ in a session, a straightforward way is to first determine the relation between q_t and q_{t-1} and then apply the corresponding principle. However, practical cases are often complicated and fuzzy. For example, it is not easy to determine whether query *"Geneva food"* specializes or is generally associated with query *"Geneva travel"*. It is very challenging to accurately classify the relations between queries and contexts.

To tackle the above problems, we explore how well the principles can adapt to all possible contexts without explicitly distinguishing the types of contexts, i.e., types of query relations. Empirically we evaluate the effectiveness of principles over all the query pairs extracted from user sessions. Each consecutive query pair $q_{t-1}q_t$ is used as a test case for a ranking principle if the principle demotes or promotes at least one search result for q_t. It is possible for one query pair to be a test case for multiple principles. Table 6 shows the number of test cases for each principle as well as the evaluation results, where "Pid" indicates the principle-id. For Principles 1, 2, and 4, the Δ values pass the significance tests at the confidence level of 0.01. Since the tests are conducted in all contexts, the results suggest that Principles 1, 2, and 4 adapt well to different types of contexts.

The Δ value for Principle 3 is negative, and it does not pass the significance test. By finer analysis on the test cases for Principle 3, we observe the following. First, Principle 3 is sensitive to query relations. We manually labeled the $1,539$ test cases for Principle 3, and found that only 55% of them are of generalization relation. As shown in Table 5, the Δ value on generalization pairs is positive. Although that positive Δ value does not pass the significance test either, it suggests that Principle 3 may perform differently on different types of relations. Second, the test cases for Principle 3 in all contexts is only about 4% in our data set. This is because the search results for the current query q_t are unlikely to contain the terms not in q_t but in the previous query q_{t-1}.

Name	Description	Pid
OrgPos*	The original position of u	-
IsClicked	Whether $u \in (U^c_{t-1} \cup \ldots U^c_1)$	1
IsSkipped	Whether $u \in (U^s_{t-1} \cup \ldots U^s_1)$	1
CosBMA($\cdot$)	Cosine(u, q_δ)	2
JacBMA($\cdot$)	Jaccard(u, q_δ)	2
CosAMB($\cdot$)	Cosine(u, q_ϵ)	3
JacAMB($\cdot$)	Jaccard(u, q_ϵ)	3
CosBAA($\cdot$)	Cosine(u, $q_\cap$)	-
JacBAA($\cdot$)	Jaccard(u, $q_\cap$)	-
CosTopics	Cosine(C_u, $C_\cap$)	4
JacTopics	Jaccard(C_u, $C_\cap$)	4

* OrgPos is only used in RankSVM-F.

Table 7: The major features in ranking models.

Finally, since the number of test cases is small, it is unclear whether Principle 3 is effective. We may ignore Principle 3 due to the small amount of applicable cases.

4. CONTEXT-AWARE RANKING

Although we have developed effective context-aware ranking principles, to achieve fully context-aware ranking in Web search practice, there are still several challenges in applying the principles. First, a user session may contain more than two queries, while we only discuss the principles formulated based on two queries. *How can we extend the applicability of the principles?* Second, given a query as well as its context, there might be multiple principles applicable. *How should we jointly execute the principles?* Third, user sessions contain rich information. Many factors, such as the positions of the documents returned by the search engine and the terms shared by the current query and the previous ones, may all be useful in ranking documents. *How can we incorporate those factors into the ranking model?*

To address the above challenges, we employ a learning-to-rank approach to build context-aware ranking models. We derive features from the ranking principles developed in Section 3 and incorporate the features into learning-to-rank models. The ranking features extend the context information from the immediately preceding query and answers to all previous queries and answers within the same session of the current query. Besides the features derived from the previous ranking principles, we also incorporate other factors mentioned above as features of the ranking models. We create training data from search sessions and train the ranking models offline. In online ranking, the trained models can carry out context-aware ranking using the available context information. It is not necessary to explicitly specify which principles to be used. By taking a learning-to-rank approach, we can address all the challenges identified above.

As a concrete example, we use RankSVM [7], a state-of-the-art learning-to-rank model to demonstrate our approach. RankSVM learns an SVM model for classification on the preference between a pair of documents. In the training stage, the RankSVM model takes as instance an ordered pair of documents with respect to a query under a context. Specifically, the i-th training example corresponds to query $q^{(i)}$ and documents $d^{(i)}_A$ & $d^{(i)}_B$ under context $c^{(i)}$, and consists of $(u^{(i)}_A, u^{(i)}_B, y^{(i)})$, where $u^{(i)}_A$ and $u^{(i)}_B$ denote the feature vectors corresponding to the two documents, respectively, and $y^{(i)}$ denotes a preference label: if $y^{(i)}$ is 1, then $u^{(i)}_A$ is preferred to $u^{(i)}_B$, otherwise, $u^{(i)}_B$ is preferred to $u^{(i)}_A$.

A feature in our RankSVM model is a function of query, document, and context. Table 7 lists the major features of the context-aware RankSVM model. Column "Pid" indicates from which principle the feature is derived. The model is flexible to combine features in addition to those from the principles (e.g., feature "OrgPos"). In Table 7, u denotes a document for query q_t. U^c_i and U^s_i denote the set of clicked and skipped documents for q_i $(1 \le i \le t-1)$, respectively. $q_\delta = q_t \setminus (q_{t-1} \cup \ldots q_1)$ and $q_\epsilon = (q_{t-1} \cup \ldots q_1) \setminus q_t$ are the differences between the current query and previous queries in the session. $q_\cap = q_t \cap (q_{t-1} \cap \ldots q_1)$ is the set of common terms among queries in the same session. $Cosine(\cdot,\cdot)$ denotes Cosine similarity, $Jaccard(\cdot,\cdot)$ denotes Jaccard Index. The common topics $C_\cap$ are derived by intersecting the topics of q_t with those of previous queries.

One issue is how to combine the original ranking of the search engine. In general, there are two possible approaches. We can use the original position of a document returned by the search engine as a feature in the RankSVM model. We denote this approach by RankSVM-F. Alternatively, we can train the RankSVM model without the original position feature. Given a test case, we combine the original ranking list R_0 from the search engine with the list R_1 from the RankSVM by Borda's ranking fusion method [1], that is,

$$score(u) = \alpha \cdot \frac{1}{R_0(u)} + (1-\alpha) \cdot \frac{1}{R_1(u)}, \qquad (1)$$

where $\alpha \in [0,1]$ is a parameter, and $R_0(u)$ and $R_1(u)$ are the positions of document u in R_0 and R_1, respectively. As a special case, when $\alpha = 0$, the Borda's fusion score completely ignores the search engine ranking. We denote this case by RankSVM-R0. Otherwise, the model is denoted by RankSVM-R1. Both RankSVM-F and RankSVM-R1 are *re-ranking* models since they incorporate search engine's ranking, while RankSVM-R0 is a *ranking* model.

5. EXPERIMENTAL RESULTS

We prepare the experimental data from a search log of a major commercial search engine. Since our ranking models use context features, we extract the search sessions with more than one query. In our search log, the percentage of such sessions is about 50%, which is consistent with the results reported by the previous studies [2, 4]. Among the 37,320 extracted sessions, we use half of them for training and validation, and the remaining half for testing.

In the following, we first describe the training process of the RankSVM models, including RankSVM-F, RankSVM-R0, and RankSVM-R1. We then compare the performance of our RankSVM models with a baseline proposed by Shen *et al.* [15] using both manually labeled data and user click data. We use the BatchUp method in [15] as the baseline since it has the best reported performance in [15]. BatchUp does not incorporate search engine's ranking. Finally, we conduct case studies and discuss the experimental results.

We train the RankSVM models from manually judged document pairs with respect to given queries and their contexts. Given a randomly selected search session with more than one query, we form an example (q_t, c, d_A, d_B), where q_t is the last query in the selected session, context c consists of the previous queries and the search results clicked on or skipped by the user before q_t in the session, and d_A and d_B are among the top five documents for q_t returned by the search engine. Please note d_A is not necessarily ranked higher than d_B by the search engine. Each document consists of its title, snippet, and URL.

For each example, a judge is asked to infer the user's search intent based on q_t as well as the context c. Then,

	RS-F	RS-R1	RS-R0	Baseline
Among 500 labeled pairs				
Num of Correct Pairs	247	239	242	203
Num of Error Pairs	109	100	113	150
Num of Unclear Pairs	144	161	145	147
P(Correct)	49.4%	47.8%	48.4%	40.6%
P(Error)	21.8%	20%	22.6%	30%
Improvement†	**27.6%***	**27.8%***	**25.8%***	**10.6%**
Over all test pairs				
Reverse Ratio on Pairs	37.6%	22.9%	40.1%	42.4%

Table 8: The performance of different methods on human-labeled data.

the judge reviews the titles and snippets of d_A and d_B and gives the preference between d_A and d_B as if he or she were the user who issued query q_t within the given context c. The judge does not know the original order of d_A and d_B returned by the search engine. The judge can choose one of the three options: 1) d_A is more preferable than d_B; 2) d_B is more preferable than d_A; and 3) unclear. A judge may choose the third option if he or she is not sure about the user's search intent, or d_A and d_B are equally preferred. In our experimental setting, each example is labeled by three judges, and we take the majority of labeled results as the ground truth. An example is "unclear" if 1) at least two judges label it as "unclear"; or 2) one judge labels "unclear", and the other two judges have inconsistent preferences.

From the judged examples, we pick $1,500$ cases which are not labeled as "unclear". We use 1,000 cases for training and the remaining 500 cases for validation. There are two parameters for our methods: C required by SVM for all the three RankSVM models and α in Equation 1 for the RankSVM-R1 model. We tune the parameters on the validation data and set $C = 1,000$ and $\alpha = 0.45$. We will use this parameter setting for all the following experiments.

Performance on manually labeled data.

We first compare on the manually labeled data the performance of the three RankSVM models, the baseline, and the search engine. For each of the four context-aware methods (RankSVM models and the baseline), we randomly select 500 examples where the method reverses the original order of d_A and d_B returned by the search engine.

Table 8 shows the results, where "RS" stands for RankSVM. We consider that a method has a correct case against the search engine if the order given by the method is consistent with the judged preference. Otherwise, the method has an error case. We calculate the percentages of correct and error cases over all the 500 reversed pairs, denoted by P(correct) and P(error) in Table 8, respectively. The row "Improvement" in Table 8 is the difference between P(correct) and P(error), which indicates how much a (re-)ranking method improves over the search engine. We also conduct significance tests on the improvements of different methods. All tests use the t-statistic and set the confidence level to 0.01.

All of the four context-aware (re-)ranking methods (three RankSVM models and the baseline method) make significant improvement over the search engine. Moreover, all of our three models, RankSVM-F, RankSVM-R0, and RankSVM-R1, perform significantly better than the baseline. This is because the baseline applies a single ranking strategy and may not adapt well to various types of contexts in Web search. Our models encode multiple principles for context-aware ranking as features and automatically adapt to different types of contexts.

We also compute the reverse ratio, i.e., the percentage of document pairs for which a (re-)ranking method reverses the orders by the search engine, over all the document pairs from the test sessions. This measure indicates how likely a method will reverse the order of a random pair of search results returned by the search engine. Table 8 shows the reverse ratio for each method. The two general ranking models, RankSVM-R0 and the baseline, have the highest reverse ratio. One of the re-ranking model, RankSVM-F, has a reverse ratio comparable with those of the two general ranking models, suggesting that the original position feature may not play a critical role in the model. The other re-ranking model, RankSVM-R1, is the most conservative. This is because we set a large α value ($\alpha = 0.45$) in the Borda's fusion method (Equation 1).

	RS-F	RS-R1	RS-R0	Baseline
MCP of Methods	2.922	2.916	2.923	2.966
MCP of Search Engine	3.096	3.096	3.096	3.096
MCP Improvement†	**0.174***	**0.180***	**0.173***	**0.130**
Reverse Ratio on Lists	68.9%	51.0%	66.1%	69.2%

† The MCP improvement of a method over the search engine is in bold if it passes the significance test, and marked with a star if it is significantly larger than that of the baseline.

Table 9: The performance of different methods on user click data.

Performance on user click data.

Although manually labeled data is usually of good quality, there could be two concerns. First, the judging process is expensive, and thus cannot be scaled up. Second, a search intent inferred by the judge may not be consistent with that of the real user. Therefore, we further evaluate the performance of the ranking methods by using real click data.

Since we consider users' clicks as their preference on search results, we only select the sessions in the test data such that the last query in a selected session must have at least one click. This results in 13,651 sessions. We follow the previous studies [6, 7, 8] and assume that 1) a user views and clicks on search results from top to bottom; and 2) a user keeps viewing search results until the one that is one position lower than the last document clicked on. For example, suppose the last URL clicked on by a user for query impression q_t is at position 5, we consider the user views all search results at positions from 1 to 6. Then, those six search results form a test case and will be (re-)ranked by our models and the baseline.

The performance of the (re-)ranking methods can be evaluated by whether they promote the search results which are clicked on by users to higher positions. Specifically, for the i-th test case, we derive the set $U_c^{(i)}$ of the clicked URLs for the last query $q_t^{(i)}$. Then, we aggregate all the test cases and calculate the *mean click position* MCP= $\frac{\sum_i \sum_{u \in U_c^{(i)}} R(u)}{\sum_i |U_c^{(i)}|}$, where $R(u)$ is the rank of u in a ranked list. A smaller MCP indicates a better ranking method.

Table 9 shows the performance of different methods on the test data, where "RS" stands for RankSVM. The row "MCP Improvement" records the differences between the search engine and the (re-)ranking methods. We also conduct significance test on the improvements. All methods have significantly lower MCPs than that of the search engine. In other words, all methods perform better than the search engine. Again, the improvement of the baseline method passes the significance test, but it is not as large as those by the RankSVM models.

We also test for each method the percentage of lists in

which the method reverses the order of at least one pair of search results. Similar to Table 8, the RankSVM-R1 method is most conservative, while the other three methods have comparable reverse ratios. For each method, the reverse ratio in Table 9 is much higher than that in Table 8. This is because, in Table 8, we consider the reverse ratio for pairs of search results, while in Table 9, we consider the reverse ratio for lists of search results. Since a list usually contains multiple pairs, the reverse possibility increases substantially.

Summary of performance tests.

We consider the human labeled data and user click data complementary to each other. For example, the human labeled data overcomes the noise and position bias in user clicks, while the user click data is large and truly reflects the preference of users. Interestingly, the experimental results on both data agree with each other on the following aspects. First, all the four context-aware methods, i.e., the three RankSVM models and the baseline, are better than the search engine. This confirms the effectiveness of context-aware ranking. Second, all our three RankSVM methods perform better than the baseline in context-aware ranking. As explained before, this is because our models consider different types of contexts in Web search. Third, RankSVM-F, RankSVM-R0, and the baseline have comparable reverse ratios, while the RankSVM-R1 method is relatively conservative due to a large α value ($\alpha = 0.45$). Finally, on both test data sets, the RankSVM-F and RankSVM-R1 methods show larger improvements than that of the RankSVM-R0 method, suggesting the usefulness of considering the original ranking of the search engine. However, the evidence is not strong enough to pass the significance test.

Case studies and discussions.

We conduct case studies on both situations where our ranking models succeed and fail. Recall the examples in Tables 1-4. For all the four examples, all of our models yield a ranking in which the documents clicked on by the user are ranked higher than those skipped by the user. However, the baseline only works well on the last example. This is because the baseline gives rise to the documents which are similar to both the current query and its context, which consists of the previous queries as well as the summaries of the previously clicked documents. In the last example, the summaries of the clicked documents for q_{t-1} contain terms about games. Consequently, the language model which incorporates the context information boosts the 4th result for q_t to the top position. In this case, the baseline bears a similar spirit with our Principle 4.

In the first three examples, the users reformulate, specialize, and generalize their initial queries in the hope to see some new results. However, in these cases, the baseline does not consider the types of contexts and still applies the single principle which tends to provide information similar to that appeared in the previous queries. Consequently, the ranking results may not meet the users' information needs well. On the contrary, our ranking models incorporate multiple ranking principles and automatically adapt to various types of contexts.

We also investigate some cases for which our models make wrong decisions. We find three major reasons for those cases. First, the ranking principles developed in Section 3.1 do not necessarily hold. For example, in some sessions, people simply click on the documents which have been just clicked before in the same session. As explained in previous studies [18], some users may use queries or keywords (such as *"msn news"*) to "bookmark" a Web page (such as `www.msnbc.msn.com`). In some other sessions, the search results do not improve much after the users refine their queries. In such cases, the users may choose to click on some search results they skipped for previous queries.

The second reason for false re-ranking may be our implementation of the ranking principles. In a real search session, the user first raised a query *"super bowl"* and then submitted query *"super bowl nine"*. Our implementation of Principle 2 promotes the search results containing the term "nine" to higher positions than that of the Wikipedia page for Super Bowl IX, which contains the term "IX" instead of "nine". In fact, the Wikipedia page is the result the user clicked on.

The remaining errors may come from our employment of RankSVM as the ranking model. Although RankSVM is one state-of-the-art ranking models, many other learning-to-rank models have been proposed in the literature [11]. It is still an open question which model is the best. Similarly, more studies are needed on what kind of models are most suitable for context-aware ranking.

6. CONCLUSIONS

In this paper, we studied the problem of using context information in ranking documents in Web search. We conducted an empirical study on real search logs and developed four principles for context-aware ranking. We further adopted a learning-to-rank approach and incorporated our principles to ranking models. The experimental results verified the effectiveness of our approach.

7. REFERENCES

[1] Borda, J.C. Mémoire sur les élections au scrution. *Histoire de l'Académie Royal des Sciences*, 1781.

[2] Cao, H., et al. Context-aware query suggestion by mining click-through and session data. In *KDD'08*, 2008.

[3] Cao, H., et al. Context-aware query classification. In *SIGIR'09*, 2009.

[4] Cao, H., et al. Towards context-aware search by learning a very large variable length hidden markov model from search logs. In *WWW'09*, 2009.

[5] Dou, Z., et al. A large-scale evaluation and analysis of personalized search strategies. In *WWW'07*, 2007.

[6] Guo, F., et al. Efficient multiple-click models in web search. In *WSDM'09*, 2009.

[7] Joachims, T. Optimizing search engines using clickthrough data. In *KDD'02*, 2002.

[8] Joachims, T., et al. Accurately interpreting clickthrough data as implicit feedback. In *SIGIR'05*, 2005.

[9] Jones, R., et al. Generating query substitutions. In *WWW'06*, 2006.

[10] Lau, T. and Horvitz, E. Patterns of search: Analyzing and modeling web query refinement. In *ICUM'99*, 1999.

[11] Liu, T.Y. *Learning to rank for information retrieval.* Foundation and Trends on Information Retrieval, Now Publishers, 2009.

[12] Qiu, F. and Cho, J. Automatic identification of user interest for personalized search. In *WWW'06*, 2006.

[13] Rieh, S.Y. and Xie, H. Patterns and sequences of multiple query reformulations in web searching: a preliminary study. In *ASIST'01*, 2001.

[14] Shen, D., et al. Q2C@UST: Our Winning Solution to Query Classification in KDD Cup 2005. *SIGKDD Explorations*, 7(2):100–110, 2005.

[15] Shen, X., et al. Context-sensitive information retrieval using implicit feedback. In *SIGIR'05*, 2005.

[16] Tan, B., et al. Mining long-term search history to improve search accuracy. In *KDD'06*, 2006.

[17] Teevan, J., et al. Personalizing search via automated analysis of interests and activities. In *SIGIR'05*, 2005.

[18] Teevan, J., et al. Information re-retrieval: Repeat queries in yahoo's logs. In *SIGIR'07*, 2007.

[19] White, R.W., et al. Studying the use of popular destinations to enhance web search interaction. In *SIGIR'07*, 2007.