Rubert: Rule-based Query Rewrite for Gemini Search

ABSTRACT

This work propose a rule-based method for query rewrite in Gemini Search (Yahoo code name for advertising sponsor search). The existing query rewrite methods use query correlation from url co-click, search co-session and ad co-bid to generate rewrites. Unable to collect enough rich co-click/co-session/co-bid is the weakness of the these methods. In contrast, the rule based rewrite does not need these correlation sources. It uses sole queries to generate rewriting knowledge. Rules are made based on expert knowledge and query analysis. The rewriter makes use of QLAS query annotation. Knowledge of all rewrite rules are extracted from QLAS results. Our experiments show rule-based results could contribute 30% coverage over the latest product version. Some rewrite rules produced higher CTR.

1. INTRODUCTION

Search advertising is the process by which ads are served when displaying the results for an Internet search query. The ads are purchased by advertising customers or agencies operating on their behalf and target specific keywords. As part of a campaign, advertisers must decide where and what keyword combinations to purchase. These are called bidterms.

The process of matching the bidterms to the query words is more or less flexible. There are three match types for advertisers to select: exact match, phrase match, and broad match. Extract match means a query is exactly as the same as a bidterm. Phrase match means part of consequent words of a query matches a bidterm. Broad match is a more flexible than the other two. If user chooses this type, the bidterm can match any query. An exact phrase match is usually considered more valuable. The advertiser is therefore willing to pay more for its impression and have it displayed at a higher rank. That translates into a higher click through rate (CTR). Broad match's CTR is not high. But this is an important revenue source. Query rewriting is mainly for broad match. While broad match has no strict restrition, the matched query should not be far from the bidterm regarding to query intent. An irrelevent rewrite to the query will result in near zero CTR and negatively impact user search experience.

Yahoo sponsor search system has implemented multiple query rewrite methods. Some are based on web search query co-clicks. If two queries click on the same landing page, the two are rewrite candidates for each other. Some are based on co-session. If two queries are issued in the same search session, they are rewrite candidates. Some are based on co-bid queries which bid on the same ads. Other methods use feature vector to represent queries. Query and rewrite are related based on similarity of feature vectors. word2vec[1] are representative methods of this type. For detailed document about existing Yahoo query rewrite methods, refer to [1].

In this work, we propose a rule based approach for query rewriting. Queries are analyzed by QLAS. Entities are extracted from QLAS interpretation results. Query rewrite is made based on entity category. Some rewrite rules are found and rewrite knowledge is extracted from queries. This rule based approach is better used for tail query rewrite because it does not require co-click/co-bid/co-session for training, which are rich primarily for head queries.

2. QLAS

QLAS is the acronym name for query linguistic analysis system. QLAS is used in Yahoo Web search for query interpretation. The system carries ou entity extraction and tagging on queries. QLAS uses conditional random fields (CRF) to generate query interpretation candidates at first. Final QLAS interpretation candidates are reranked by a scoring function, which is a decision tree based machine learned model. QLAS has used a large tagset. It can achieve state-of-the-art performance over such tag set.

QLAS's tag set has two levels: coarse level and fine level. Coarse level has 20 categories and Fine level has 200 categories. The coarse level includes categories such as PlaceName, OrganizationName, BusinessName, EventName, PersonName, etc. Fine categories are defined under coarse categories. Fine categories are used to re-clarify coarse category. For example, for a travel related business name, fine categories can be travel/lodging, travel/airline, travel/airport, travel/packages, travel/others etc.

QLAS provides much more rich query analysis than entity extraction and tagging. It can output query vertical intent, entity attributes (color, currency, etc), entity wikipedia id linking to a wikipedia page, geo id for places, etc. QLAS also output detailed model scores such as interpretation score, query segmentation confidence score, and query intent detection scores. In this work, we only use one of the functions: entity extraction and tagging. As an example, QLAS's output of a query, 'best hotel san francisco', is that 'hotel' is a BusinessName entity, 'san francisco' is an entity with coarse category equal to PlaceName and fine category equal to 'city'. 'best' is a token, not an entity. Rubert rewrite system is based on QLAS output to extract entity related

[1]https://code.google.com/p/word2vec/

knowledge. For this query, 'hotel' is a BusinessName and will be used as a rewrite word for queries containing PlaceName.

INPUT:	best hotel san francisco
OUTPUT:	[best]_token
	[hotel]_BusinessName
	[san francisco]_PlaceName/City

3. REWRITE RULES

Basically, query rewrite is to change a query's original word sequence to a new word sequence. If we exclude word order change, the new sequence is different in no more than three ways: some words added, some words deleted, some words substituted. There have been some published literatures targeting to model deletion and substitution [2, 3]. For example, from "san francisco" to "hotel san francisco" adds words. From "hotel san francisco" to "san francisco" deletes words. From "hotel san francisco" to "accomodation san francisco" substitutes words. According to this logic, if we can find patterns dealing with adding words, deleting words and substituting words, we can use the patterns for query rewrite. After observing travel queries, our first observation is business related words more likely appearing with a place name entity. For example, these are queries related to 'san francisco':

san francisco hotel
san francisco accomodation
san francisco car rental
san francisco travel
san francisco restaurants

From these queries, we can generate a rule, PlaceAddBusiness = PlaceName+BusinessName words. If a query has a place name, we can rewrite the query by adding a business name keyword. By using similar method we define the following rules:

1. PlaceRemove: If a query contains PlaceName entity, remove it. For example, "hotel san francisco" will be "hotel".

2. OrgRemovePlace: If a query contains both OrganizationName and PlaceName, remove PlaceName. For example, "21c hotel cincinnati ohio" will be "21c hotel". This rule is a subset of the first rule.

3. PlaceAddBusiness: If a query contains PlaceName, add business related contextual words. We extracted entity related contextual words. This step will be described next section. An example is from "2016 alaska cruise" to "2016 alaska cruise all inclusive package".

4. OrgAddProduct: If a query contains OrganizationName entity, add product contextual words. Organization makes products. Apple makes Iphone; Amazon makes Kindle. When a query contains an organization, this rule adds product words. Product words extraction will be described next section. An example is, "hertz rental" to "hertz rental minivan".

5. BrandAddProduct: Same as above. If a query contains BrandName entity, then add product contextual words. QLAS distinguish BrandName and OrganizationName. For the same word, Apple, it is BrandName for query "Apple iphone"; but "OrganizationName" for query "Apple computer". For example, "toyota" rewrites to "toyota camery".

6. EntityAddTokens: most queries consist of entities and tokens(non-entity words). We extract tokens and use them as rewrite candidate. Tokens belong to the query's entity tokens. This rule is to add these tokens based on the query's entity. For example, "hotel" is rewritten to "best hotel". 'best' is a non-entity word belonging to the business token category.

To formalize the above rules, what we are doing is to find optimal W using query expansion model to maximize $P(add(W)|W \ni Q)$, or deletion model to maximize $P(delete(W)|W \in Q)$. Next section describes our method to find W. More examples are shown on Table 2.

4. CONTEXTUAL WORD EXTRACTION

All the rules we defined in last section use three type of contextual words: business related entity, product related entity and entity related tokens. Business entities are used in Rule 3. Product entities are used in Rule 4, and 5. Entity related tokens are used in Rule 6. We used QLAS to process all the 240K travel queries. We used QLAS to identify BusinessName entity and ProductName entity. We then extract BusinessName entities and ProductName entities from the query. We aggregated and sorted the entities. In the end, we extracted total 1.8K unique BusinessName entities and 130 unique ProductName entities. The number of ProductName entities are far less than BusinessName entities probally because we are using travel queries. ProductName entities are not frequent for travel queries. Some top frequent examples of BusinessName entities and ProductName entities are shown in the first two columns of Table 1.

While we extract entities for Rule 3-5, we extract tokens for Rule 6. We only extracted tokens from single-entity queries. These tokens are contextual words of the query. Tokens' entity is the query's entity. For example, we extract 'best' from query 'best hotel'. Because 'hotel' is a BusinessName entity, 'best' is a BusinessToken (BusinessName tokens). In rewriting, if a query contains a single BusinessName entity like 'restaurants', its rewrite will be 'best restaurants'.

From 240K travel queries, we extracted the following entity's tokens. The number are the total number of tokens for each entity.

PlaceToken	2,669
BusinessToken	2,476
OrganizationToken	1,613
ProductToken	154
PersonToken	64

Table 1 shows top 10 extracted contextual words. PersonName entity is not popular in travel queries. We extracted the least number of PersonTokens. Some query examples are 'britney spears tickets', 'celine dion tickets'.

5. EVALUATION:BUCKET TEST

We extracted BusinessName entities, ProductName entities and entity-related tokens. We applied Rule 1-6 by adding or deleting these entities and tokens to 240K travel queries. Total 250M rewrites are generated before joining with bid terms. Our first try is to product an offline query

Table 1: Top 10 contextual words for rule type

Business entity	Product entity	Place token	Business token	Org token	Product token	Person token
hotels	cars	things do	cheap	deals	rent	tickets
hotel	sandals	vacation packages	deals	reservations	rental	vegas
flights	suv	map	best	tickets	cheap rental	vegas tickets
motels	passenger van	vacation	coupons	reviews	rent cheap	2014
car rental	car	vacations	discount	packages	travel	concert
car rentals	trucks	extended stay	online	all inclusive	renting	discount tickets
cruises	minivan	attractions	cheapest	coupons	cheap rent	show
rental cars	kayak	travel	best deals	vacation packages	cheap	last supper
lodging	cargo van	places stay	prices	2015	rent full size	concert tickets
resorts	van	weather	last minute	locations	rent large	tour

Table 2: Rubert query rewrite examples

Rule type	Query	Rewrite
PlaceRemove	south mountain resort lincoln nh	south mountain resort
	hotel minerva rome	hotel minerva
	thunderbird lodge lake tahoe	thunderbird lodge
	cleveland car rental	car rental
OrgRemovePlace	hotel minerva rome	hotel minerva
	thunderbird lodge lake tahoe	thunderbird lodge
	holiday inn kennesaw ga	holiday inn
	circle line cruise new york	circle line cruise
PlaceAddBusiness	legoland	legoland hotels
	aruba	all inclusive aruba
	edmond oklahoma	edmond oklahoma hotel
	punta cana	punta cana resorts
OrgAddProduct	travelocity	travelocity cars
	mykonos	mykonos cruise
	hertz rental	hertz rental cars
BrandAddProduct	windstar	windstar cruise
AddPlaceToken	nepal	nepal holiday
	maui	maui tourism
	nadi	nadi travel
	coors field	coors field hotels
AddOrgToken	priceline	priceline travel
	enterprise	enterprise rent
	us air	us air airlines
	delta	delta reservation
AddBusinessToken	cruises	cruises deal
	travel agencies	travel agencies online
	rental car	rental car specials
	international flights	international flights cheap
AddPersonToken	rod stewart	rod stewart tickets
	olivia newton john	olivia newton john tickets
	celine dion	celine dion vegas

Table 3: Online metrics

Rule	clicks	impression	CTR	CPC
PlaceRemove	3	25	0.12	0.107
OrgRemovePlace	2	7	0.28	0.05
PlaceAddBusiness	1928	63705	0.038	0.27
OrgAddProduct	27	641	0.04	0.22
AddPlaceToken	2286	64315	0.036	0.20
AddOrgToken	2418	43743	0.056	0.13
AddBusinessToken	7	168	0.042	0.22
AddPersonToken	14	247	0.056	0.10
AllTable	28577	633821	0.045	0.26

rewrite table. We can join with bid terms. Thus we can remove the unused rewrites which is not bidded by advertisers. Keep rewrites matched with bid terms. After join, 3M rewrites are remained. Therefore, only 1% rewrites are used. One benefit of joining is improving query rewrite relevance. Many irrelevant Rubert rewrite are removed.

We start a bucket test in June. But we only used rule PlaceAddBusiness because other rules are not ready at the time of bucket test. There are 1.7M rewrite generated by rule PlaceAddBusiness. The PlaceAddBusiness rewrites combining with rewrites generated by other methods such as coclick/co-session/co-bid merged into a new query rewrite table. We pushed this new query rewrite table into bucket test. When user issues a query on Yahoo search engine, rewrites of the query are extracted from the table. If a bid term match the rewrite, the ads of the bid term will be selected. If the ads is also passed verifications from the following modules such as ad quality, ad click predict model and ad targeting model, the ad will be shown on Yahoo search result page. We have an impression count. Users can see the ads. They may click the ads or skip it. If it is clicked, we have a click count. User behavior log data are recorded and stored into a dedicated data source for Gemini search, Kunefe[2]. Kunefe saves not only click and impression, also includes adgroup id, rewrites, match type, cost per click(cpc), north first click, etc. Some query rewrite examples from Kunefe are shown on Table 2.

We used Kunefe July data to calculate metrics. The metrics we used are clicks, impression, CTR, and averaged CPC. CTR equals to total clicks divided by total impressions. CPC is averaged by total sum of single click's CPC devided by all clicks. CTR can evaluate query rewrite relevance. The better relevance, the higher CTR. CPC can evaluate how much Yahoo earns for a single click.

From Kunefe we extracted only query and rewrite that Rubert rewrite have. We can get these rewrites' clicks, impression, CPC and rule type. The metrics are listed in Table 3.

Rule PlaceAddBusiness, AddPlaceToken and AddOrgToken have higher impressions. For travel queries, place related rules like PlaceAddBusiness and AddPlaceToken should have higher impression. Low impression of PlaceRemove may be due to lower relevance score and will not be chosen by click predict model. AddOrgToken has higher impression may be due to many OrganizatonName in travel queries. We also observed Rubert's CTR was not high.

Table 4: Ad group coverage

#(adgroup) range	#(query)
[0,10)	19471
[10,20)	11930
[20,30)	8748
[30,40)	7368
[40,50)	8027
[50,60)	8779
[60+)	132391

The last row, AllTable, includes other type of rewrites like coclick/cosession/co-bid, etc. Comparing AllTable results, Rubert has lower CTR. But CTR of Rule AddOrgToken is higher than AllTable results. Out of all rule types, AddOrgToken gives the best results. The impression of Rubert is about 30% of that of AllTable.

Ads coverage is another evaluation metric for query rewrite. In sponsor search, advertisers can define ad groups. Each adgroup contains an ad and several related bid terms. it is not a good rewrite if all rewrites fall into one ad group, because only one ad can be selected. A good query rewrite can cover many ad groups. We can use ad group coverage to see if a query rewrite can generate more diversified ads. Table 4 shows statistics of adgroup coverage. The first column is adgroup number range. The second is number of queries. As the first row, there are 19,471 queries have rewrites that can trigger less than 10 ad groups. About 132,391 (55%) queries can generate rewrites that match more than 60 ad groups.

6. CONCLUSIONS

This work evaluates a new query rewrite, Rubert. This rewrite can contribute 30% of impressions in bucket test. Some rules achieved higher than average CTR. In future work, we want to expand Rubert to more entity category. Current work only generate business and product word list. Similar methods can expand to other entity type even fine QLAS categories. This work's source code is downloadable from, https://git.corp.yahoo.com/ruiqiang/rubert .

7. REFERENCES

[1] Yahoo internal gemini guru talk 2015, https://docs.google.com/ spreadsheets/d/ 1b7mxua0rivzwq_kongrrldye5g_qqi7yc4ruqku17us/ edit#gid=0.

[2] R. Jones and D. C. Fain. Query word deletion prediction. In *SIGIR '03*.

[3] R. Jones, B. Rey, O. Madani, and W. Greiner. Generating query substitutions. In *WWW'07*.

[2]https://docs.google.com/document/d/1s_evoCTulZU3WEhmBfhf86NRXcXNUuSKV09EJ3-BbJI/edit