Web Intelligence and Agent Systems: An International Journal 0 (2017) 1–0
IOS Press

Using Online Data Sources to Make Query Suggestions for Children

Pera Maria Soledad [a] and Ng Yiu-Kai [b,*]
[a] *Department of Computer Science, Boise State University, Boise, Idaho 83725, U.S.A.*
E-mail: solepera@boisestate.edu
[b] *Computer Science Department, Brigham Young University, Provo, Utah 84602, U.S.A.*
E-mail: ng@compsci.byu.edu

Abstract. Existing popular web search engines have been widely used for retrieving information of interests by their users and offer query suggestions (QS) to assist them in exploring the wealth of information online. These search tools, however, are designed without any specific group of users in mind and thus are not tailored towards the specific needs of children, which can diminish their usability and design objectives when they are employed by children. Given the increasing use of the Web for educational and entertainment purposes by children, there is an urgent need to help them search the Web effectively. In this paper, we present a QS module, denoted CQS, which assists children in finding appropriate query keywords to capture their information needs by (i) analyzing content written for/by children, (ii) examining phrases and other metadata extracted from reputable (children's) websites, and (iii) using a supervised learning approach to rank suggestions that are appealing to children. CQS offers suggestions with vocabulary that can be comprehended by children and with topics of interest to them. We conducted a number of empirical studies using keyword queries initiated by children, besides gathering feedback on the usefulness of CQS-generated suggestions through crowdsourcing. The performance evaluation of CQS revealed the effectiveness of the methodology of CQS. In addition, it demonstrated that CQS-generated suggestions were preferred over suggestions provided by Bing and Yahoo! and at least as comparable to queries suggested by Google.

Keywords: Query suggestion, children, backpropagation, phrase generation, probabilistic models

1. Introduction

Children regularly use search engines as the starting point in their quest for online information [29]. Unfortunately, their search experiences can be negatively influenced by their lack of skill in formulating adequate search queries. While query suggestion (QS) modules designed for widely-used search engines facilitate query creation for a general audience, they were never designed from a child's perspective. (The query suggestions generated by known web search engines for two sample queries created by Utah's elementary school children, as shown in Table 4, illustrate that suggestions made by known search engines do not necessarily target children's interests and/or information needs.) A survey conducted by McAfee, a software security company, in 2013 shows that children spend an average of 6.5 hours a day online[1], making them active web users. Furthermore, as reported by Gossen [12], the percentage of children who use the Internet "increases with age from 21% by six years old to 98% by thirteen years old," that explains why Google plans to create a children version of its search engine[2]. With the growth of this segment of Internet users, there is a demand for the design and develop-

*Corresponding author. E-mail: ng@compsci.byu.edu

[1] http://www.mcafee.com/us/resources/reports/rp-digital-deception-survey.pdf
[2] http://www.usatoday.com/story/tech/2014/12/03/google-products-revamped-for-under-13-crowd/19803447/

ment of QS modules specifically tailored towards children.

Query suggestions made by existing general-purpose QS modules may require advanced reading level on complex topics which children have difficulty understanding and appreciating them [5]. The discrepancies between children's and adults' search behaviors/interests were further verified by the study conducted by Bilal & Kirby [4] and another one by Torres et al. [27] who analyzed AOL query logs and the DMOZ kids/teens directory and have identified significant differences between the set of commonly-created queries for a general audience and queries that seek information on children's content, such as average length of queries (3.2 words for children versus 2.5 for regular users). Even though search engines designed specially for children, such as Safesearchkids.com, Kidsclick.org, and Kidrex.org, exist, majority of them are not equipped with a QS module. To aid children with their quest for information that satisfy their needs, we have developed CQS[3], a QS module that offers query suggestions for children who are individuals of age up to twelve-year old. CQS is the result of the study conducted to analyze the validity of exploiting resources explicitly targeting children (from content written to and for children, to hierarchy of categories that captures concepts/themes of interest to children) to create suggestions that are diverse, free from adult-based bias, and appealing to children. These suggestions capture the intended information needs of children.

Existing query recommendation/transformation techniques attempt to improve a submitted keyword query through word replacements, insertions, and deletions [7]. CQS, on the other hand, minimizes the effort required by a child in specifying his/her search intent by providing query recommendations, which are N-gram suggestions, that yield the suffix to the user's initial keyword query. Note that suggestions made by CQS for a child's query Q, which include Q as prefix, follow the common design methodology of existing web search engines, such as Google, Yahoo!, and Bing. Instead of reformulating Q, all of these popular engines offer suggestions by appending keywords to the end of Q[4]. The same applies to children search engines, including Kidzsearch.com, a leading kids' safe search engine.

CQS relies on bigrams extracted from multiple reputable websites (as discussed in Section 3) that include content written for or by children, and is different from existing QS modules targeting children [28, 29] which rely on tags assigned by adults to describe children/teenager's websites for making query suggestions. CQS also considers bigrams extracted from Simple.Wikipedia.com, denoted SimpWiki, which is a small, evolving archived collection of documents written in basic English. SimpWiki targets young readers and adults learning English as a second language. Based on the content, which includes simple vocabulary and is written so that children can understand, CQS can suggest meaningful and useful phrases as queries to children.

2. Related Work

Query Suggestion in itself is a non-trivial task for web search engine designers, since it requires disambiguating user's search intent using very few query keywords, i.e., 2.8 words on the average [14]. If a QS module is designed for addressing children's information needs, as opposed to a general, i.e., more mature, audience, then it has to analyze children's search intents and behaviors, which are different from those of adults [9]. In fact, children struggle with limited vocabulary to pose their queries to search engines and have difficulty in issuing appropriate search keywords [8].

While research on QS systems targeting children is limited, research work on QS systems for a general audience is rich and well-documented. Existing QS approaches for a general audience [23] either adopt probabilistic methodologies, examine query logs, apply strategies based on random walks, consider concepts/categories, i.e., subject areas of interest, or rely on ontologies, to name a few.

In suggesting queries for young audiences, Torres et al. [28] introduce a QS module based on tags created at Delicious.com. The team constructs a bipartite graph using tags and their corresponding URLs, and suggests queries as a result of a random walk on the bipartite graph that is biased towards children's content. Later research [29] presents further enhancements based on topical and language modeling fea-

[3] An earlier version of CQS, which includes the initial design and assessment of CQS, was published in [26].

[4] We examined hundreds of suggestions made by Google, Yahoo!, and Bing, and all of them were generated by a completion-based approach.

tures, such as topic-sensitive Page Rank and children-related vocabulary distribution, to more effectively suggest queries for children. Similar to the approaches described above, CQS does not rely on query logs to generate suggestions. CQS, however, differs from these QS modules for children, since, instead of using a bipartite graph, CQS considers diverse features that aim to precisely capture children's intents. More importantly, CQS relies on content written for/by children to suggest queries as opposed to relying on tags that are often provided by adults and may be poorly defined due to the lack of quality control on user tags and thus can be inherently noisy [7].

Eickhoff et al. [10] present a two-step query expansion strategy for children. Given a query Q, it retrieves top-n results in response to Q from various search engines and uses tags assigned to each retrieved web page at Delicious as keywords to expand Q. In addition, the name of high-level semantic categories (inferred from Wikipedia and the DMOZ.org taxonomy) associated with these tags are treated as expansion terms as well, which are non-stop keywords that can be preceded by a sequence of *connection words*[5]. While CQS generates cohesive phrases to guide users in formulating queries that capture their information needs, the approach in [10] simply provides tag-related terms to add to the given query to locate children-related content.

To improve children's web searches, Gyllstrom et al. [13] develop a link-based algorithm which finds web pages for children. The algorithm suggests web pages that include simple vocabulary terms using SimpWiki. While CQS is not designed to search for web pages for children, we share a similar design methodology, i.e., providing simple suggestions that are suitable for kids. Inspired by using SimpWiki, we also extract some of the keywords for suggestions from SimpWiki.

Karimi et al. [17] also focus on facilitating information discovery tasks for children by offering query suggestions. Similar to their proposed strategy, CQS also considers specific children vocabulary and popularity of terms among children texts. However, the QS strategy in [17] depends on Ubersuggest[6] to identify candidate suggestions, which limits the type of suggestions that can be generated. CQS, on the other hand, relies on n-grams to generate suggestions on-the-fly.

Vidinly and Ozcan [30] adopt a query-log based strategy to provide QS for K-12 students. Their approach examines query logs to identify potentials suggestions, which are then re-ranked based on a score computed by aggregating information pertaining to educational features, grade similarity, log session information, and path frequency algorithms. CQS, unlike its counterpart presented on [30] is not constrained by the existence of query logs created by children.

An initial assessment on children search engines that specifically target children was conducted by us. A number of these search engines do not offer query suggestions, which include Cyber Sleuth Kids[7] and Dib Dab Doo[8]). Even though children engines such as Kidz Search[9], Safe Search Kids[10] and Sweet Search[11] offer query suggestions, these search engines are not necessarily tailored towards children. For example, consider the query "school" posted on the Kidzsearch engine. In response to the query, suggestions made by Kidzsearch include *school uniform debate* and *school uniform statistics*, neither of which is of particular interest to children.

Instead of considering simple bigram and n-gram phrases for query suggestions as CQS does, Yuefeng et al. [20] and Zhong et al. [31] apply text mining approaches to identify useful features and discover effective patterns in text documents, respectively. Relevance features are extracted from text documents in [20] using both positive and negative patterns, whereas effective patterns are mined from text documents in [31] by discovering specificities of patterns based on term distributions in the extracted patterns. Both of these approaches can be adopted by CQS for phrase construction during the children query suggestion process. The tradeoff between using simple n-gram frequencies adopted by CQS instead of discovering text patterns is that the former is simple and yet effective to accomplish the task of suggesting children queries,

[5] A *connection word* [2] is either a preposition, a conjunction, or an article, which is treated as a stopword and is *not* counted as words in a suggestion but is retained to capture the precise meaning of a suggestion.

[6] Ubersuggest.org
[7] http://cybersleuth-kids.com/
[8] http://www.dibdabdoo.com/
[9] http://www.kidzsearch.com/
[10] http://www.safesearchkids.com
[11] http://www.sweetsearch.com/

whereas the latter is more sophisticated but requires the adoption of more complicated text mining techniques.

3. Our Query Suggestion Methodology

Using bigrams extracted from readerviewskids.com, dogonews.com, stonesoup.com, timeforkids, and cmlibrary.org, etc. which are children's websites, and well-established probabilistic/information retrieval models, CQS identifies each *candidate suggestion* for a user query Q and the closest *categories* to which Q belong. Table 1 shows the list of categories defined at well-known children's websites and considered by CQS, which include stonesoup.com, dogonews.com, and kids.nationalgeographic.com. All the websites, from where various types of information extracted by CQS for generating query suggestions, are shown in Table 2.

For each candidate suggestion CS, CQS computes its *ranking score* using a backpropagation (BP) model as presented in [21] on a number of features that capture the (i) *likelihood* of CS, which identifies the search intent of an individual user based on the pre-defined category c to which Q belongs, (ii) *N-gram frequency* of CS based on the co-occurrence of bigrams in children's documents in c, (iii) *vocabulary support* of CS, which is a suitability (i.e., kid-friendliness) measure of the keywords in CS for children, (iv) *simplicity of phrase keywords* in CS, which determines the comprehensiveness of keywords in CS for children, (v) *distribution of phrase keywords in children's documents*, which measures the likelihood of generating CS from documents addressing children content belonged, (vi) *locality* of CS, which computes the degree of co-occurrence of bigrams in CS that are also in children's documents and reflects the degree of cohesiveness of keywords in CS, and (vii) children's *subject headings*, which denote concepts, events, or names employed by librarians to categorize and index children's books according to their themes. Phrases with top-ranked scores, which are simple and easy to read and better capture topics of interest to children, are offered as suggestions for Q.

The overall process in making query suggestions by CQS for any child's query is shown in Figure 1.

3.1. Candidate Suggestions

To determine the candidate suggestions for query Q with m (≥ 1) words, i.e., terms, CQS examines the *frequency of occurrence* of words that follow the last word t_m in Q in a category c. CQS identifies the frequencies of the top-five most frequently-occurred t_{m+1} words following t_m, denoted $f(t_m, t_{m+1})$, in c. For each one of the top-five words t_{m+1} in c, CQS considers the next top-five $f(t_{m+1}, t_{m+2})$ frequency values and so on to determine candidate suggestions in different categories for Q. Given that suggestions including seldom-occurring words are less likely to make it to the top ranking positions among the suggestions for Q, CQS considers only the top-five most frequent words. In doing so, CQS speeds up its processing time without affecting its accuracy, a fact that has been empirically verified and discussed in [25].

To obtain the frequency distribution of bigrams that are used in generating candidate suggestions, CQS examines the consecutive word occurrences in the 82,000 documents belonged to the 16 categories extracted from children's websites (see Table 2). Using word occurrences, CQS considers $f(t_m, t_{m+1})$ and creates phrases as suffixes of Q, which yield candidate suggestions for Q.

3.2. Category Likelihood

Given a query Q, CQS computes the likelihood of keyword(s) in Q matching the contents of different categories. To determine the *category likelihood* of Q, CQS employs the multinomial model, along with the well-known Bayes' rule[12], to compare the probability distribution of terms in different categories using the m (≥ 1) terms in Q as shown below.

$$P(c|Q) = \frac{\prod_{i=1}^{m} P(k_i|c)P(c)}{\sum_{c \in C} \prod_{i=1}^{m} P(k_i|C=c)P(C=c)} \quad (1)$$

where C is the set of 16 pre-defined categories considered by CQS, $P(c)$ is the probability of observing $c \in C$, which is the ratio of documents in c to the total number of documents used to train the multinomial model, and $P(k_i|c)$ is the probability that the i^{th} term in Q is observed in c as determined using the multinomial model defined below.

$$P(k|c) = \frac{tf_{k,c} + 1}{|c| + |V|} \quad (2)$$

[12] A detailed description on how to apply the popular Baye's rule for category likelihood's estimations can be found in [7].

Table 1

Categories defined and used by CQS based on information available at children websites, such as Dogonews.com and Kids.nationalgeographic.com

Categories			
(1) Adventure	(2) Animals	(3) Books	(4) Comedy
(5) Did You Know	(6) Education	(7) Entertainment	(8) Health
(9) History	(10) Music	(11) Nature	(12) Science
(13) Space	(14) Sports	(15) Video	(16) World

Table 2

Websites used by CQS for generating query suggestions for children

Website	URL	Data Used by CQS
Spaghetti Book Club	www.spaghettibookclub.org/	Training phrases for BP
Good Book Recommendations	best-kids-books.com/good-book-recommendations.html	Training phrases for BP
Mother Daughter & Son Book Reviews	motherdaughterbookreviews.com/	Training phrases for BP
American Literature: The Children's Library	americanliterature.com/childrens-library	Bigrams
Reader Views: reviews, by kids, for kids	readerviewskids.com/reviews-by-age/	Bigrams
Dogo news: Fodder for young minds	www.dogonews.com/	Bigrams, Categories info. & likelihood, Naïve Bayes (NB) feature (kid class)
Time for Kids	timeforkids.com	Bigrams, NB feature (kid class)
Kidworld	www2.bconnex.net/∼kidworld./	Bigrams, NB feature (kid class)
National Geographic: Kids	kids.nationalgeographic.com/kids/	Bigrams, Categories info., NB feature (kid class)
Simple English Wikipedia	Simple.Wikipedia.org	Bigrams, Phrase simplicity
Stone Soup: The Magazine by Young Writers and Artists	www.stonesoup.com/archive/stories	Bigrams, Categories info., Category likelihood, NB feature (kid class)
BookHive: Your Guide To Children's Literature	www.cmlibrary.org/bookhive/books/	Bigrams, Categories info., Category likelihood
Reading Rockets	www.readingrockets.org/article/22366	Children's vocabulary
BigIQkids	bigiqkids.com/SpellingVocabulary/Lessons/wordlist SpellingFirstGrade.shtml	Children's vocabulary
The Game Gal	http://www.thegamegal.com/printables/	Children's vocabulary
Children's Library	archive.org/details/iacl	NB feature (kid class)
Free Kids Books	freekidsbooks.org/	Bigrams
Mighty Books	mightybooks.com/	Category likelihood
Poetry for Kids	www.poetry4Kids.com	Bigrams
Gutenberg - Children's Fiction	gutenberg.org/wiki/Children's_Literature_(Bookshelf)	Bigrams
Randomly Selected 10,900 Wiki Documents	www.wikipedia.org/	NB feature (generic class)

where $|c|$ is the number of non-stop, stemmed keywords in the training documents of c, and $|V|$ is the number of distinct non-stop, stemmed keywords in the 41,847 training documents extracted from children's websites, and $tf_{k,c}$ is the *frequency of occurrence* of term k in c. Each candidate suggestion CS generated from c is assigned a *category likelihood value* of $P(c|Q)$ which determines the likelihood of CS being from c.

CQS treats the likelihood value computed in Equation 1 as one of the measures to determine the significance of CS and computes the *category likelihood* score of CS with respect to c, denoted $CL(CS, c)$, as

$$CL(CS, c) = P(c|Q). \quad (3)$$

If the probability of keywords in Q belonged to category c is *high*, then a candidate suggestion originated from c is treated as a *more promising* suggestion for Q. CQS offers diverse query suggestions by considering various categories to which a given *ambiguous* query[13] can be interpreted, which occur often.

[13] A query is *ambiguous* if it has several possible meanings or interpretations. For example, the user who creates the keyword query

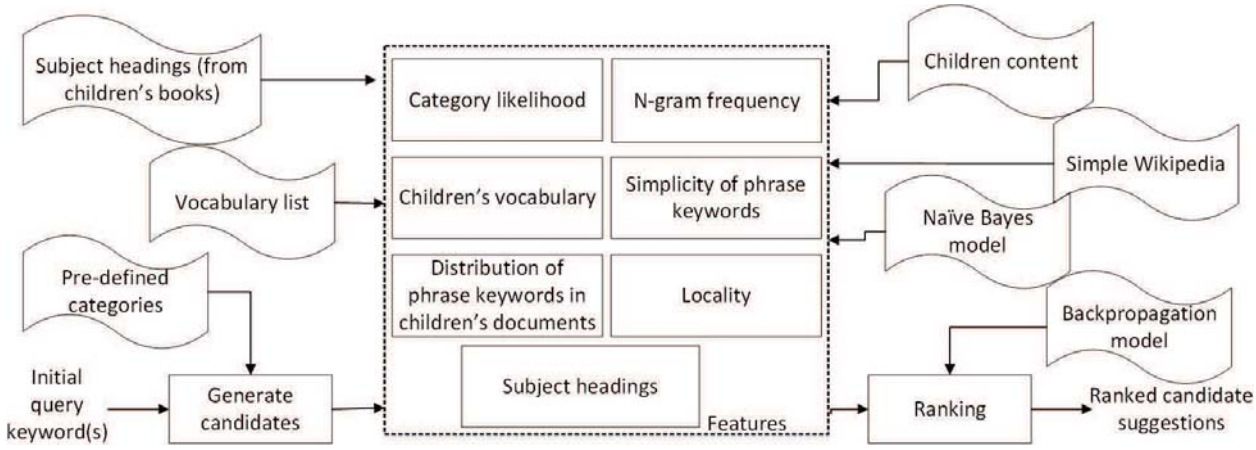

Fig. 1. The query suggestion process of CQS

Candidate suggestions with recommended keywords in the same category as the original keyword in a given query are given preference by CQS.

3.3. Bigram and N-gram Frequencies

CQS relies on frequency distribution of bigrams within categorized documents to provide statistics of consecutive term occurrences in different categories, i.e., categories shown in Table 1. Recall that we define a *term* as a non-stopword keyword, which can be preceded by a sequence of *connection words*. For example, if a user enters the query keyword "information," a suggestion "information about animals" makes more sense than the suggestion "information animals," since in the latter case the relationship between the two keywords is missing. To obtain the frequency distribution of bigrams, CQS examines the consecutive term occurrences in the aforementioned 82,000 children's documents (including SimpWiki documents), which are distributed across 16 categories.

We consider *bigram* frequency distribution of terms so that searching for related terms to a user query becomes *more efficient* than examining the frequencies of occurrence of sequences of more than two terms which increases the database size [2] for the document collection and significantly impacts the search time for terms related to a given query.

For a given query Q with the only keyword A, CQS generates candidate suggestions for Q by considering phrases, which are N-grams that includes A, such as ABC and $ABCD$ in which AB, BC, and CD are bigrams belonging to documents of the same category c. In generating phrases, CQS first considers all the frequent bigrams with the leading A. Based on the statistical data as to the *frequency of occurrence* of words B that follow A, and words that follow B, and so on, CQS concatenates the bigrams such that the 2^{nd} word in the preceding bigram is the 1^{st} word in the subsequent bigram to generate candidate suggestions. Thus, candidate suggestions are N-grams generated from across different categories, which are constructed based on the frequencies of word co-occurrences. CQS computes the N-Gram frequency score for each candidate suggestion CS.

Given a CS, which is a sequence of words, $t_1, \ldots, t_n$ $(n > 1)$ extracted from documents in c, CQS computes the average (N-Gram) frequency of the sequence below.

$$f(t_1, t_2, \ldots, t_n) = \frac{f(t_1, t_2) + f(t_2, t_3) + \ldots + f(t_{n-1}, t_n)}{n - 1} \quad (4)$$

where $f(t_{i-1}, t_i)$, $1 < i \leq n$, is the frequency of the bigram made up by the word t_{i-1} followed by word t_i. A high N-gram frequency value of CS indicates that, in general, CS includes highly co-occurring bigrams in c and is treated as a favorable suggestion.

We consider a special case $f(t_{i-1}, \text{EOS})$, where EOS stands for "End Of Sentence". $f(t_{i-1}, \text{EOS})$ captures the frequency with which term t_{i-1} is the *last*

"apple" may ask for information about Apple computers or the fruit apple.

term in a sentence. CQS considers all the terms following t_{i-1}, including EOS[14].

CQS applies N-gram frequency to determine which candidate suggestions should be considered favorable.

3.4. Children's Vocabulary (CV)

CQS determines the "children-friendliness" of the vocabulary used in a candidate suggestion by consulting a *vocabulary dictionary* comprised of words appropriate for children that were downloaded from children's word lists posted at a number of children's websites, including, but not limited to Reading Rockets, Big IQ Kids, Game Gal (see Table 2 for details), as well as Library of Congress (Children's Subheading). If a term in a suggestion is found in the dictionary, it is assigned the value of 1; otherwise, it is given the value of 0. For a candidate suggestion CS with multiple terms, CQS *averages* the values over all the terms in CS and obtains a single value between 0 and 1, which is called the *Vocabulary score* for CS. A Vocabulary score of value closer to 1 indicates that the corresponding suggestion is child-friendly.

3.5. Simplicity of Phrase Keywords

As indicated earlier, one of the design goals of CQS is to offer query suggestions that children can understand, i.e., providing simple query suggestions. To determine the *simplicity* of a candidate suggestion CS, CQS measures not only whether non-stop words in CS appear in children's vocabulary (CV), but also how often they are seen in web pages that are written in a simple style. Given that texts consisting of short sentences and simple words are deemed easier to read than those including longer sentences and rare words [1], it is assumed that web pages written using *basic* English vocabulary and *shorter* sentences are tailored for children. SimpWiki is such a website that includes these web documents. Hence, we maintain a count of all the words in the entire collection of documents archived at SimpWiki.

A highly-ranked candidate suggestion CS can be partially affected by having keywords of CS in the SimpWiki documents. CQS considers the *normalized frequencies* for the occurrences of keywords in SimpWiki documents, with values between 0 and 1. A value closer to 1 for a non-stop keyword in a phrase indicates that the word is very often found in SimpWiki, which reflects its *degree of simplicity*. For example, the term "call" with a value 0.7 indicates that it is very commonly found in simple text documents, i.e., kid's literature, as compared with "torrent", which is assigned a value 0.000327, is less-frequently-used in kid's simple text documents. The values of the keywords in CS are *averaged*, and the averaged value is referred as the *simplicity score* of CS.

Simplicity vocabulary SV generated from SimpWiki differs from the children's vocabulary CV introduced in Section 3.4. First, words in CV are *not* extracted from a collection of text documents. Instead, they are words that a child is expected to know. Second, a word in SV is simple but may be absent in CV. In SV, we include words, such as "Nintendo" (a video game console), "Transformer" (the name of a toy), or "Anna" (a character in a Disney movie), which are common terms of children's folk culture, or terms that children are exposed to on a regular basis through mass-media, i.e., pop-culture. (See [22] for a detailed discussion on children's exposure to terms/concepts addressing children's folk, mass-media, and commercial cultures.) Terms in SV are *simple* based on their *frequency of exposure* to children. However, they are not commonly-occurred words in children's literature, and hence are not part of, i.e., included in, CV. Quite often to distinguish which one of the two candidate suggestions with all the words included in CV is more appealing to a kid, CQS must rely on their *simplicity* scores.

3.6. Distribution of Phrase Keywords in Children's Documents

While *simplicity of phrase keywords* indicates how often keywords in a candidate suggestion CS are used by kids, it does not show whether the words are more likely to be found in documents pertaining to kids than in documents belonging to generic audience. To measure whether the word distribution of CS within kids' documents is more likely than in a general-audience's documents, CQS determines its *Naïve Bayes (NB) classification score* using the NB model. This score captures the probability distribution of keywords in CS that are also in *children's* content. We trained a *multinomial model* as presented in [7] using 10,900

[14]Technically, EOS is not a term according to its definition. However, we consider the frequence of $f(\text{EOS}|t_i)$ as an exceptional case in CQS.

documents randomly chosen from Wikipedia.org for the non-kid's class and 25,115 documents from children's websites, such as Dogo News, National Geographic for Kids, Time for Kids, etc. (see Table 2 for further details), for the kid's class. Although the number of documents in kid's class is more than the number in non-kid's class, it does not create an imbalance, since documents in the former are comparatively shorter than the Wikipedia documents, resulting in a vocabulary smaller than the non-kid's class. Given CS, the NB feature calculates the likelihood of CS using the Naïve Bayes' rule.

3.7. Locality

By concatenating different bigrams in a category into an N-gram phrase, some undesirable phrases, such as "greek language french cyclist" which consists of frequent bigrams 'greek language', 'language french', and 'french cyclist', can be created and should be avoided. To eliminate their creations, CQS determines the *locality* score for each candidate suggestion CS. The locality score, as defined below, which is based on the *Lennon Similarity measure* [19], captures the *likelihood* of all the bigrams in CS being extracted from the same document(s) within a given category c, such that the *smaller* the number of documents in c in which all the bigrams in CS occur, the *less likely* CS is an appealing one.

$$locality(CS, c) = \frac{S_n}{Min\{S_{l_1 l_2} - S_n, \ldots, S_{l_{n-1} l_n} - S_n\} + S_n} \quad (5)$$

where n is the number of terms in CS, l_i $(1 \leq i \leq n)$ is a term in CS, S_n is the number of documents in c that include all the bigrams in CS, and $S_{l_i l_j}$ $(i, j > 0)$ is the number of documents in c that include bigram $l_i l_j$. It is easy to see that the *more* bigrams in CS, the *fewer* the number of documents that include all the bigrams in CS which yields the *lower* the locality score. By using the *Min* function in Equation 5, we normalize the locality score of CS without penalizing CS based on its length.

3.8. Subject Headings

It has been shown [11] that searching information on the Web can be facilitated by searching for the topics of the desired information. With that in mind, we have designed CQS to examine the topics of information addressed in candidate suggestions and penalize suggestions that are associated with topics/themes that are not commonly associated with children content. To accomplish this task, CQS relies on *Library of Congress Subject Headings* (LCSH), which is a de facto universal controlled vocabulary and constitutes the largest general indexing vocabulary in the English language. LCSH, which are keywords or phrases that denote concepts, events, or names, are employed by librarians to categorize and index books according to their themes, i.e., topics. Examples of LCSH include "Fairy tales" and "Fear of the dark-Fiction".

To identify, among the large number of LCSH, the subject headings that address topics of interest to children, we (i) examined LCSH assigned to 30,000 randomly selected books known to be suitable for children, which are with readability levels between the K-6 grades defined by publishers and (ii) generated a list of 10,749 children's LCSH, denoted $cLCSH$, that describe text content in children's literature using subject keywords.

In examining the topical information of a candidate suggestion CS, CQS employs Equation 6 to determine the *degree of closeness* of CS and children's subject headings, denoted $SHScore$. To compute the SHScore feature score of CS, CQS compares CS against each subject heading SH in $cLCSH$, and chooses the *highest* similarity value between CS and the subject headings as the value that quantifies the degree to which CS addresses themes suitable for children. The degree of similarity between CS and SH is computed using the *word correlation factor* (wcf)[15]

[15] wcf reflects the *degree of similarity* between any two words based on their (i) *frequencies of co-occurrence* and (ii) relative *distances* in a collection of Wikipedia documents. CQS relies on wcf, as opposed to WordNet-based similarity measures, since it has been empirically verified that the former correlates with human assessments on word similarity more accurately than the latter [24].

[24] of each (non-stop, stemmed) word in CS with respect to (non-stop, stemmed) words in SH[16].

$$SHScore(SH, CS) = MAX_{SH \in cLCSH} \frac{\sum_{i=1}^{n} Min\{\sum_{j=1}^{m} wcf(CS_i, SH_j), 1\}}{n} \quad (6)$$

where n (m, respectively) is the number of distinct (non-stop, stemmed) words in CS (SH, respectively), CS_i (SH_j, respectively) is a (non-stop, stemmed) word in CS (SH, respectively), and $wcf(CS_i, SH_j)$ is the correlation factor of CS_i and SH_j.

The *Min* function in Equation 6 imposes a constraint on summing up the correlation factors of words in the description of CS and SH. Even if a word in the description of CS (i) matches exactly one of the words in SH and (ii) is similar to some of the remaining words in SH, which yields a value greater than 1.0, CQS limits the sum of their similarity measure to 1.0, which is the word-correlation factor of an exact match. This constraint ensures that if CS contains a dominant word w in its description which is highly similar to a *few* words in SH, w alone cannot dictate the content resemblance value of CS with respect to SH. Words in SH that are similar to most of the words in CS should yield a greater $SHScore$ value than the $SHScore$ value of words in SH that are similar to only one dominant word in CS. The *Max* function, on the other hand, ensures that the $SHScore$ of CS reflects the highest similarity of CS among all the subject headings which most effectively captures the topics/theme of CS.

A candidate suggestion CS with a *high* $SHScore$ illustrates that CS is closely related to contents in children's literature and hence is treated by CQS as *more favorable* compared with other suggestions with *lower* $SHScore$.

Example 1 Consider a candidate suggestion "british movies", which is generated by CQS in response to the user query "british." In comparing against the subject headings *digestion*, *finance person*, *television*, and *films* using word-correlation factors, the corresponding degrees of resemblance with respect to "british movies" are assigned the SHScore of 0, 0, 2.7 $\times 10^{-7}$, and 3.8 $\times 10^{-7}$, respectively. Based on the aforementioned resemblance scores, *films* is the closest to the suggestion. By using the subject heading feature, CQS assigns 3.8 $\times 10^{-7}$, which is the similarity score computed with respect to *films*, as the SHScore value of "british movies".

Now consider the four subject headings again for another candidate suggestion "british pancakes", which is assigned the value 0 as its SHScore, indicating that there is no suitable topic description for the suggestion based on the set of subject headings. Furthermore, in considering the two suggestions "british movies" and "british sitcoms" against the subject heading *television*, CQS assigns higher SHScore to "british sitcoms" than "british movies", i.e., 1.1 $\times 10^{-6}$ versus 2.7 $\times 10^{-7}$, since *television* is more accurately reflecting the subject area of "british sitcoms" than "british movies." □

3.9. Ranking Candidate Queries

Using the individual scores of the features introduced earlier, which are computed for each candidate suggestion CS, CQS ranks the candidate suggestions belonged to multiple categories so that the top-k suggestions[17] are recommended to its user by CQS. CQS relies on a backpropagation model to generate a single score for each candidate suggestion CS that reflects the cumulative effect of each of the seven features computed for CS and determines the degree to which CS is a suggestion suitable for children. BP is a machine learning algorithm based on neural networks, which learns weights associated with different inputs, i.e., features in our case, and is often used to perform categorization and/or ranking tasks.

In training the BP model for CQS, 138,579 training instances were used. Each training instance includes a given noun-phrase, in lieu of a query, and is associated with the seven different feature scores computed for the corresponding noun-phrase and a label, which is either 1 or 0, to designate whether the

[16] While the list of subject headings is created by indexers and librarians, each of the headings in cLCSH addresses a topic/theme of interest to children. Using the word-correlation factors, we measure the degree to which a candidate suggestion CS refers to various topics/themes of interest to children, without requiring the terms in CS to exactly match the terms in $cLCSH$, which are known to be defined adults who are indexers or librarians. In doing so, we explicitly ensure that CS remains child-friendly.

[17] K in top-k suggestions is determined by the software developer who implements CQS and is recommended to be in the range of 4 and 10.

noun-phrase is a children or generic query, respectively. In gathering the training instances of children queries, noun-phrases from a number of children websites, including spaghettibookclub.org, motherdaughterbookreviews.com, and best-kids-books.com,[18] were extracted. Training instances associated with generic queries, on the other hand, included queries extracted from the AOL query log[19], a well-known source of general-audience queries.

4. Experimental Results

In this section, we present the results of the empirical studies conducted to assess the design of CQS and compare its performance with well-known QS modules.

4.1. Dataset and Metrics

Due to the lack of benchmark datasets to evaluate the design methodology and performance of QS modules for children, we turned to a number of 7- to 12-year-old children who are 1^{st}- to 6^{th}-grade students at a local school. During the month of April 2014, we asked the students to first create keyword queries that they would like to use to conduct their searches. We collected 127 children queries. Children were then provided with a number of CQS-, Google-, Yahoo!-, and Bing-generated suggestions to evaluate. We selected these popular search engines for baseline purposes, as opposed to search engines specifically targeting children, since it has been reported that children rank Google, Yahoo!, and Bing as their three most favored engines for conducting their daily information discovery tasks [3].

We used *eight*, five unigram and three bigram, queries randomly chosen out of the (keywords in) 127 unique queries provided by 127 elementary school students, who are 3^{rd} to 6^{th} graders, to evaluate the performance of CQS. (Table 3 shows a few of the 127 queries offered by the children.) Altogether, 25 students in 3^{rd} grade, 36 in 4^{th} grade, 26 in 5^{th} grade, and 40 in 6^{th} grade participated in the empirical study. For each query Q, we applied CQS to generate the top-4 query suggestions, which were mixed with the top-4 suggestions of Q offered by Google, Yahoo!, and Bing. Note that we assessed the performance of CQS against Google, Yahoo!, and Bing, since the latter are widely-used search engines. Furthermore, the top-4 suggestions of CQS were used, since Google offers four suggestions per query, the least number of generated suggestions among the QS modules considered for comparison purpose.

Table 3
Sample queries created by children

Children's Queries	Grade	Children's Queries	Grade
hawaiian	3	tallest person	3
migration route	4	youtube	4
san antonio spurs	5	animals with no fur	5
yahoo mail	6	cute kitten pics	6

Each child who participated in the study was asked to choose *four* useful suggestions for each test query. Table 4 shows two of the test queries and their corresponding suggestions used in the evaluation, and the remaining queries are *arctic circle*, *chocolate chip*, *football*, *greek*, *information*, and *snow*. The top-4 most frequently chosen suggestions for each test query Q, among the choices provided by the 43 children who participated in the evaluation, were treated as the *gold standard* of Q.

We acknowledge that the number of queries considered for the evaluation of CQS by school children is relatively small. However, given that (i) evaluations involving children are difficult to conduct due to privacy constraints [28] and (ii) we only had access to students for a limited amount of time—each student involved in the assessment was given 15 minutes to complete the evaluation, which was imposed by their school administrators—we were forced to limit the number of queries to be assessed to eight which allowed each student to spend an average of at most 2 minutes on evaluating suggestions for a query. Had we been given access to more students at the school and/or more time to conduct our evaluation, we would have compiled the results based on more than eight queries.

We applied a simple *counting* scheme to evaluate the query suggestions made by CQS. For each CQS-offered query suggestion that was chosen by a child (as a useful suggestion), a *point* is rewarded for the suggestion. Using this counting strategy, we consider the top-4 counts of suggestions for each one of the eight test queries, which yields the *gold standard* for our

[18] These sources include diverse content for creating sample children queries addressing multiple topics.

[19] http://goo.gl/TOIcz5

Table 4

Two sample queries and suggestions recommended by Google, Yahoo!, Bing, and CQS, respectively. Highlighted queries are among the top-4 gold standard suggestions for the respective query

Query	Google	Yahoo!	Bing	CQS
ice cream	ice cream clothing	ice cream maker	ice cream sandwich cake	**ice cream pie**
	ice cream	ice cream recipes	ice cream recipes	ice cream month
	ice cream shoes	ice cream sandwich cakes	ice cream maker	ice cream time
	ice cream cake	ice cream cake recipe	ice cream maker recipes	**ice cream cone**
british	**british**	british open leaderboard 2014	**british airways**	**british people**
	british airways	british open 2014	british open 2014	**british television series**
	british slang	british open	british airways flight status	british india
	british virgin islands	**british airways**	british museum	british actress

evaluation. A total of 43 children participated in the evaluation.

To determine the effectiveness of CQS and existing QS modules (considered for comparison purpose) in making useful suggestions to children, we have computed the *Normalized Discounted Cumulative Gain (nDCG)* value [15] on their corresponding top-4 suggestions for each test query. nDCG *penalizes* relevant suggestions that are ranked *lower* in the list of suggested queries.

4.2. Performance Evaluation

To verify the correctness of the design of CQS we conducted a number of studies using the dataset and metrics introduced in the previous section. We first validated the correctness of selecting Backpropagation as a combination strategy. Thereafter,

we evaluated the effectiveness of each individual feature (as discussed in Sections 3.2 through 3.8) in making good query suggestions to children, which validates its usefulness. Lastly, we assessed the overall performance of CQS based on combining individual features as a whole (as defined in Section 3.9), which we compared with the performance of a number of existing QS modules adopted by popular search engines. As previously stated, we also used the nDCG metric to quantify the performance of CQS in making suggestions suitable for children against Google, Yahoo!, and Bing, being used these days[20].

[20]In computing nDCG scores, a candidate suggestion CS for a test query Q is considered a useful suggestion, i.e., relevant, if it matches any of the suggestions included in the gold standard of Q.

4.2.1. Validating CQS's Combination Strategy

CQS considers the BP model, a supervised learning-to-rank approach as presented in Section 3.9. To demonstrate the correctness of selecting such a strategy to combine the seven different scores generated by CQS for each candidate suggestion into a single *ranking* score, we compared the use of BP with two other combination strategies: (i) CombMNZ [18], which is a linear combination measure frequently used in fusion experiments [6] and (ii) Reciprocal Rank Fusion (RRF) [6].

CombMNZ, as defined in Equation 7, is applied to consider multiple existing lists of rankings on CS in CQS to determine a joint ranking of CS, a task known as rank aggregation or data fusion.

$$CombMNZ_{CS} = \sum_{f=1}^{N} CS^f \times |CS^f > 0| \quad (7)$$

where N is the number of (ranked lists of) features to be fused, which is *seven* in our case, CS^f is the normalized score of CS on the ranked list of feature f, and $|CS^f > 0|$ is the number of non-zero, normalized scores of CS in the ranked lists to be fused.

Prior to computing the ranking score of CS, it is necessary to transform the original scores in each individual ranked list of CS into a common range, which can be accomplished by applying Equation 8 to each score in each ranked list so that it is within the range $[0, 1]$, a commonly-used.

$$CS^f = \frac{S^{CS} - CS^f_{min}}{CS^f_{max} - CS^f_{min}} \quad (8)$$

where S^{CS} is the score of the feature f of CS prior to be normalized, CS^f_{max} (CS^f_{min}, respectively) is the maximum and (minimum, respectively) score in the ranked list of f, and CS^f is the normalized score for CS in the ranked list of f.

Reciprocal Rank Fusion (RRF) [6] first sorts the score list of each feature of all the candidate suggestions, which yields a ranked list of candidate suggestions corresponding to every feature in descending order. Given a list of M candidate suggestions, RRF in Equation 9 is applied to generate a single rank score for CS.

$$RRF_{score}(CS \in M) = \sum_{f \in N} \frac{1}{rank(CS^f)} \qquad (9)$$

where N is a set of features such that their scores for a particular CS are to be combined and $rank(CS^f)$ is the ranking position of CS in the ranked list of feature f.

It has been empirically verified that BP is significantly better than CombMNZ and RRF in terms of combining different features of a candidate suggestion as shown in Figure 2. The figure shows the performance of using different combination strategies on the *eight* queries that were evaluated by CQS. While the unsupervised combination strategies of CombMNZ and RRF are *simple*, their overall nDCG values are significantly lower than the overall nDCG value achieved by BP. Although BP requires training to learn the feature weights, the training process is *one-time*.

4.2.2. Feature Evaluation

To determine which feature(s) of CQS, as presented Sections 3.2 through 3.8, contribute(s) the most in making children suggestions, we relied on a test dataset, denoted *TestData*. TestData consists of 12,000 labeled instances, which include phrases and their corresponding scores computed for each of CQS features. These phrases are uniformly distributed among children/non-children categories and are disjoint from the instances presented in Section 3.9. Each instance in TestData comes with the scores computed based on each of the aforementioned features. We analyzed the capability of each (group of) feature(s) in distinguishing (non-)children phrases, which are potential candidate queries.

We computed the nDCG score of each feature using the TestData dataset in addition to the overall nDCG score of CQS computed using backpropagation as a combination strategy (as discussed in Section 3.9). As reflected by the nDCG scores in Figure 3, each individual feature underperforms the combined features used by CQS. By combining all the features, CQS takes the advantage of their individual strengths and greatly improves the degree of relevance and suitability of its generated query suggestions for children. The overall nDCG of CQS as shown in Figure 3, which is 0.51, is a statistically significant improvement ($p < 0.001$) over the nDCG score achieved by any single feature.

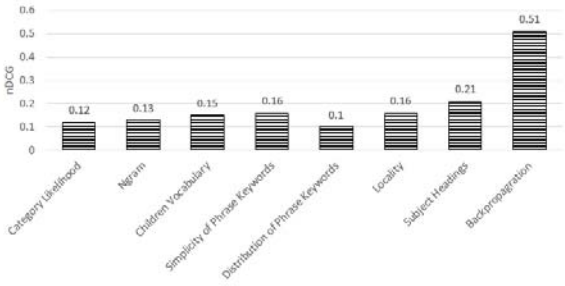

Fig. 3. Performance evaluation of (each of the features of) CQS

4.2.3. CQS versus QS Modules

We compared CQS with the QS modules employed by Google, Yahoo!, and Bing in terms of nDCG, which is an evaluation framework similar to the one adopted by the authors of [28,29]. One of the strengths of our evaluation strategy lies on the fact that we rely on children's assessments, and there is no room for adult-based bias. This is because we use keyword queries initiated by children as test queries and the top-4 suggestions selected as the gold standard are the ones chosen by children.

In *seven* out of eight queries, CQS provides a suggestion that made it either to the *first* or *second* position in the gold standard, whereas Google achieves only six out of the eight queries. Figure 4 shows the performance of Google, Yahoo!, Bing, and CQS using the nDCG measure. The results have verified that suggestions made by CQS are more appealing to children than the ones offered by Google, Yahoo!, and Bing. Although the results of CQS are not statistically significant ($p \leq 0.05$) compared with Google, they are statistically significant ($p \leq 0.01$) compared with Yahoo! and Bing based on the Wilcoxon signed-ranked test.

Besides analyzing the overall performance of CQS, Google, Yahoo!, and Bing using nDCG, we also examined their performance at the *query level*.

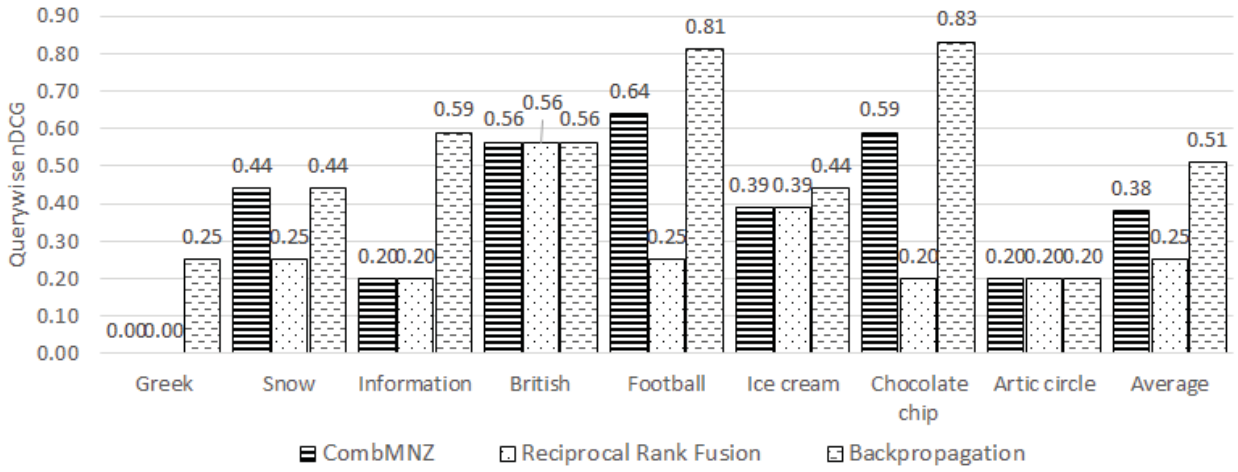

Fig. 2. Performance evaluation of three feature combination strategies in generating query suggestions for children based on the eight test queries

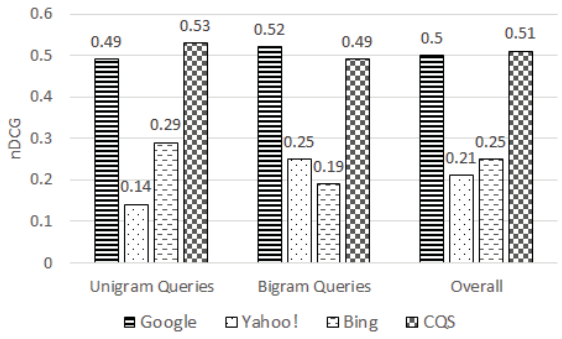

Fig. 4. The nDCG scores for Google, Yahoo!, Bing, and CQS respectively determined using their top-4 suggestions against the gold standards

As shown in Figure 5, CQS outperforms Google in making suggestions in *four* out of the 8 test queries based on the gold standard. More importantly, CQS-offered suggestions are placed at *higher* ranking positions compared to Google. In fact, among all the test queries, while CQS and Google both have equal number of suggestions posted at rank position 1, the former has more suggestions placed at positions 2 and 3. Similar to the overall performance, CQS and Google distance themselves from Yahoo! and Bing in terms of achieving higher nDCG values at the query level.

We attempted to compare CQS against other children QS modules [10,28,29]. Unfortunately, implementing these modules requires setting up different parameters which are not explicitly articulated in [10, 28,29]. For example, the authors of [28,29] create the random walk graphs, which include a foreground and background model, based on tags and URLs. Given that the (i) specific URLs considered for creating such graphs were not made available, and (ii) authors' cleaned tags and URLs to a large extent using tag normalization and tag filtering are not described in details in the respective paper, it is not possible to regenerate the foreground and the background model for comparison purpose. Furthermore, datasets presented in [10,28,29] are not available to the research community. For this reason, fair comparisons between CQS and these children QS modules are not possible.

4.3. Mechanical Turk's Evaluation

As previously stated, there are no benchmark datasets that can be used to assess the performance of QS modules for children. For this reason, we turned to Mechanical Turk[21] to conduct empirical studies that allow us to further evaluate the performance of CQS. We relied on Amazon's Mechanical Turk, since it is a "marketplace for work that requires human intelligence", which allows individuals or businesses to programmatically access thousands of diverse, on-demand workers and has been used in the past to collect user feedback on various information retrieval tasks. The performance evaluation of CQS based on independent appraisers are presented in Sections 4.3.1 through 4.3.3.

4.3.1. Relevance of CQS-generated Suggestions

We conducted a survey on Mechanical Turk in which we asked appraisers to examine a set of *ten* test

[21] https://www.mturk.com/mturk/welcome

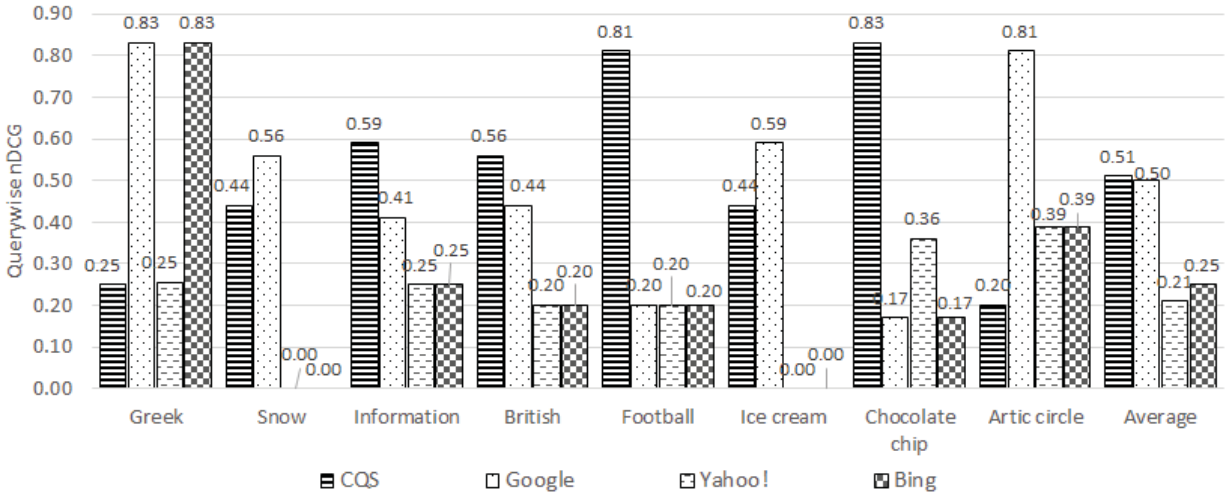

Fig. 5. Per-query distribution of nDCG scores for CQS, Google, Yahoo!, and Bing, respectively

queries and their corresponding suggestions created by CQS. For each query Q, appraisers were required to identify, among a provided set of *four* suggestions generated by CQS for Q, the ones (if any) that were suitable and relevant for children[22]. (A sample evaluation form is shown in Figure 6.)

While the ten test queries (with five queries in each evaluation form) included in the survey, which are "Disney", "Lego", "Pet", "Transformers", "National football", "Art", "Dog", "Minecraft", "Video", and "Basketball player", were selected among the query set introduced in Section 4.1 that address varied topics of interests for children at diverse school grade levels, the corresponding suggestions were generated using CQS. The goal of this survey is to quantify the degree to which queries suggested by CQS are appealing to children (from the adults' points of view). We collected 90 responses during the month of July 2014. Based on the feedback collected through Mechanical Turk, we have observed that, on the average, (close to) 50% of the recommendations generated by CQS were deemed suitable for children. (For the evaluation of the performance of CQS on each test query, see Figure 7.)

We are aware that each Mechanical Turk appraiser must be over 18 years old. We solicited appraisers of all walks of life and assessed the performance of CQS by separating the opinions of appraisers known to be educators or parents of young children[23], who have a more direct knowledge on the interests/preferences of children in terms of selecting suitable query suggestions, from the opinions of general appraisers. As shown in Figure 7, the accuracy ratios computed based on parents/educators' responses yield not only a statistically significant improvement ($p < 0.05$) over the ones based on the responses of the general appraisers, which is determined using the Wilcoxon signed-ranked test, but also confirm the appropriateness of the design methodology of CQS, i.e., offering useful query suggestions to children, since in general parents/educators are in better position to judge the usefulness of queries suggested to children than the general public.

4.3.2. Evaluations on QS Modules

We also turned to Mechanical Turk to validate our claim that queries suggested by CQS for children are more favorable than the ones generated by Google, Yahoo!, and Bing. To verify this claim, we conducted another survey on Mechanical Turk (see a sample eval-

[22]Note that the responses identified as *relevant* provided by an appraiser are treated as the gold standard for accuracy and reciprocal rank assessment, and the reported overall accuracy and MRR are based on the average of the corresponding accuracy and MRR calculated according to each appraiser's response. In other words, the suggested queries treated as "suitable and relevant" by each appraiser were not combined into a *single* gold standard for evaluation purpose, since we would like to preserve varied opinions on relevance in evaluating the performance of CQS.

[23]Mechanical Turk appraisers were asked to voluntarily answer a question which inquired whether they were parents/educators. Overall, 57% of the 90 appraisers who assessed the performance of CQS were parents/educators.

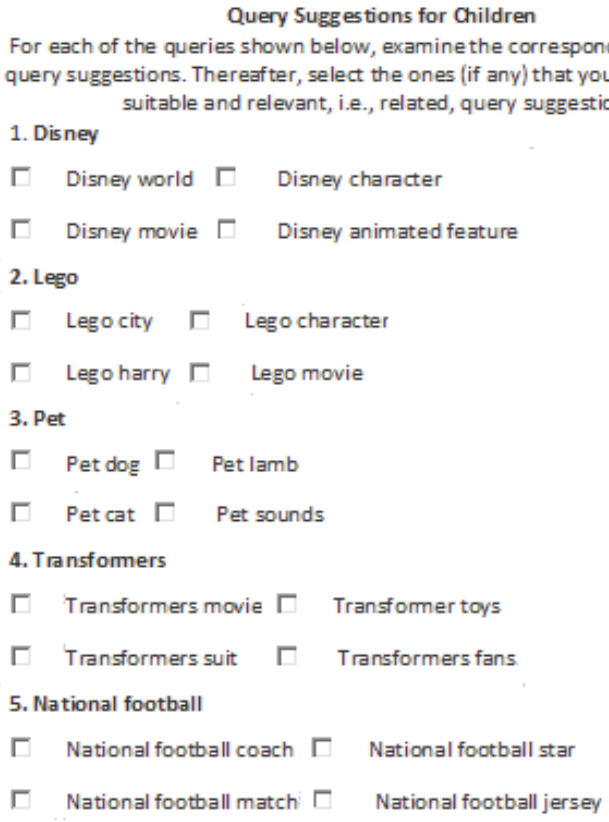

Fig. 6. An evaluation conducted by Mechanical Turk appraisers who assessed the relevance of the suggestions generated by CQS

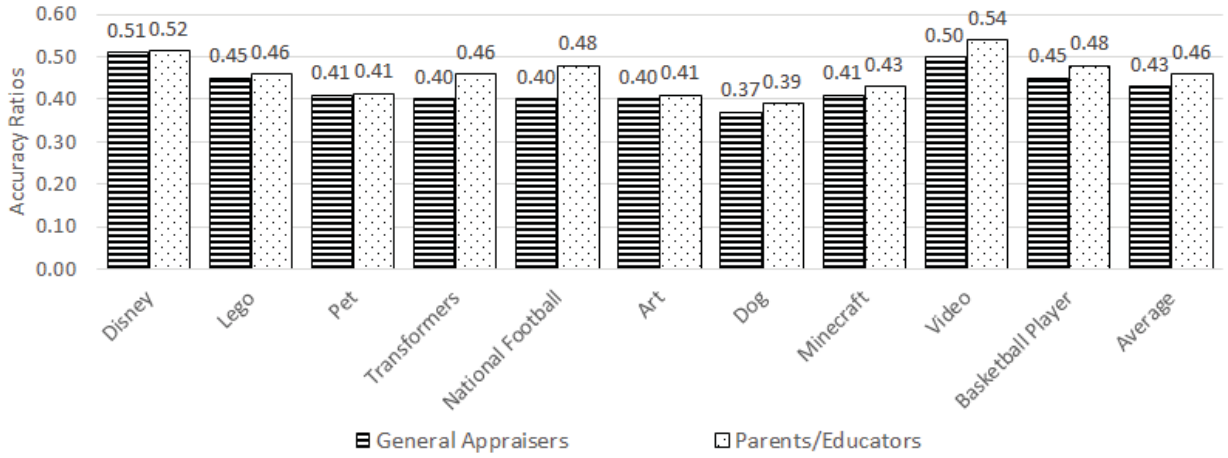

Fig. 7. Performance evaluation of CQS based on the responses of Mechanical Turk appraisers

uation form designed for this empirical study in Figure 8). The survey requested the appraisers to identify for each test query the *two* suggestions that, to the best of their knowledge, were most suitable for children. The test queries in this survey are the same queries as presented in Section 4.3.1, and the corresponding suggestions are the top-2 suggestions generated by CQS, Bing, Google, and Yahoo!. (Note that

due to overlapped suggestions offered by the four QS modules, there are less than eight suggestions for each of the test queries.) We treated the two suggestions chosen for each test query Q by each appraiser as the *gold standard* for Q. Based on the chosen suggestions, we computed the *accuracy ratio* and *Mean Reciprocal Rank* (MRR). While the former quantifies the proportion of relevant suggestions generated by a QS module, the latter computes the average ranking position of the first relevant suggestion provided by the corresponding QS module.

The accuracy and MRR scores computed according to the 90 responses collected during the month of July of 2014 are shown in Figure 9. The results, which are statistically significant with $p < 0.05$, show that appraisers often preferred children query suggestions provided by CQS over the suggestions created by Google, Yahoo!, or Bing. These findings are consistent among the 57% of appraisers who were either educators or parents of young children.

To further validate the performance of CQS, Google, Yahoo!, and Bing, in terms of their ability to generate suggestions appealing to younger audiences, we conducted another survey to gather feedback through Mechanical Turk. Using the same experimental framework described earlier in this section, we created new HITs using 28 new queries (see the new queries, along with the 10 test queries used in the July 2014 survey, in Table 5), which are selected among the ones described in Section 4.1, and vary in terms of length.

We collected 657 responses on the new HITs during the months of January and February of 2017. Based on these responses, we computed the accuracy ratio and MRR of CQS, Google, Yahoo!, and Bing, respectively. As shown in Figure 10, query suggestions generated by CQS continued to be favored by appraisers over those generated by Google, Yahoo!, or Bing, and CQS performs significantly better than Google, Yahoo!, and Bing (with $p < 0.05$) based on the Wilcoxon signed-ranked test. The results, which remain comparable among the 46% Mechanical Turk appraisers who reported being parents of young children or educators, further demonstrate the validity of the experimental results reported earlier in this section.

Note that Table 5 shows the child-formulated queries used in all of the Mechanical Turk experiments discussed in this section. For each test query, we also include the top suggestion generated by CQS and each of the web search engines considered for comparison purpose.

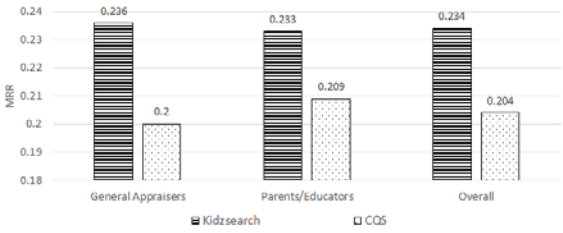

Fig. 11. Performance assessment in terms of MRR for CQS and Kidzsearch, based on Mechanical Turk appraisers' responses

4.3.3. The Need for Suggestions Tailored to Children

As we previously stated in Section 4.1, the objective of our empirical assessment is to demonstrate the need for query suggestion modules, alike to CQS, that can be used to complement search engines favored by children, such as Google. In order to further illustrate the suitability of CQS in generating query suggestions to meet the information needs of younger audiences, we conducted another experiment. In the new online study, we examined the performance of CQS and Kidzsearch, a leading kids search engine as discussed earlier.

Following the framework discussed Section 4.3, we created HITs in which we included suggestions generated by both Kidzsearch and CQS for diverse sets of unigram, brigram, and trigram queries, with a total of 28 queries (see the queries in Table 5). We asked Mechanical Turk appraisers to select, according to their interpreted intent of each query Q, the top-2 suggestions most relevant to Q. We collected 687 responses, out of which 344 were provided by appraisers who reported being either parents of young children or educators.

Based on the responses collected on Mechanical Turk, we computed the MRR and accuracy ratios. As shown in Figure 11, Kidzsearch and CQS achieve comparable performance in terms of *MRR*, i.e., the difference between their MRR ratios are not significant. The reported results are also consistent among appraisers that are parents or educators. Even though the overall differences in the *accuracy* ratios between Kidzsearch and CQS are statistically significant ($p < 0.001$) for general appraisers (as depicted in Figure 12), CQS and Kidzsearch perform almost identically according to the responses collected among appraisers who are parents/educators. Moreover, the overall accuracy ratios achieved by CQS and Kidzsearch are not statistically significant.

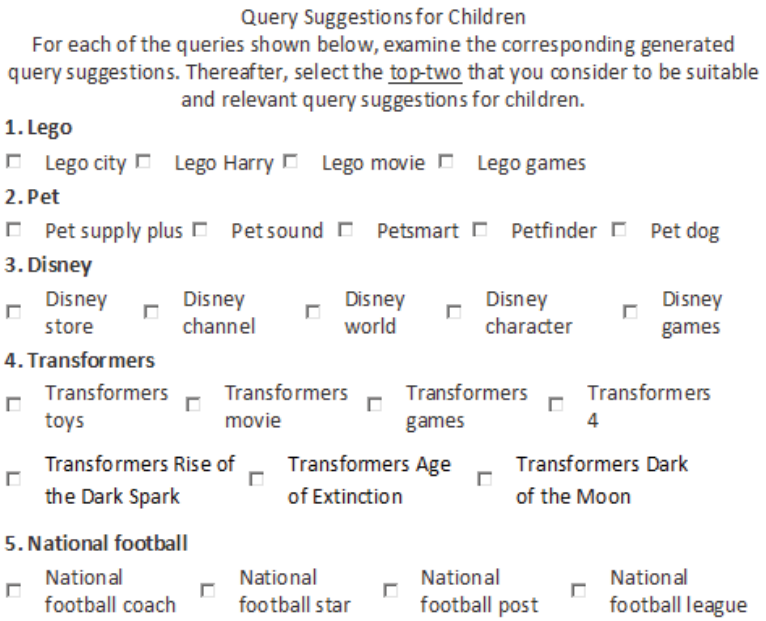

Fig. 8. The Mechanical Turk evaluation form for performance comparison of CQS and other QS modules

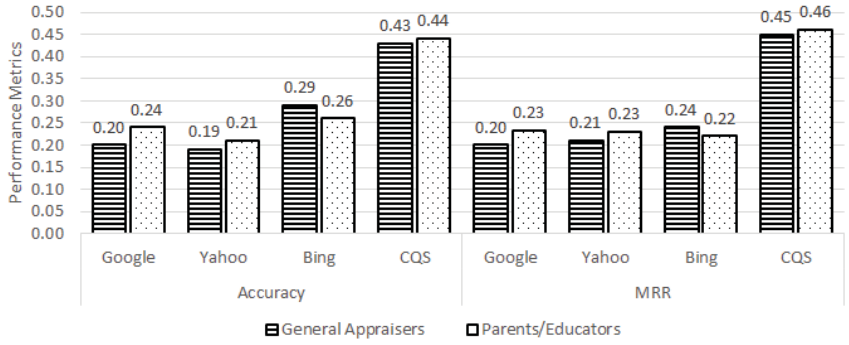

Fig. 9. Evaluations of CQS and other QS modules based on Mechanical Turk appraisers' responses

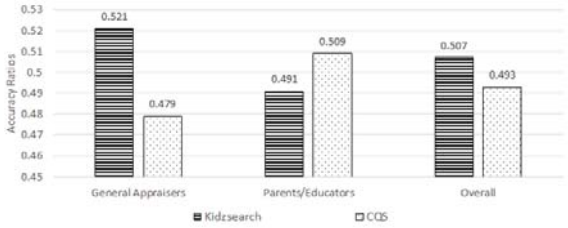

Fig. 12. Accuracy ratios for CQS and Kidzsearch based on Mechanical Turk appraisers' responses

4.3.4. Observations

In Section 4.2 and Sections 4.3.1 through 4.3.3, we presented the results of the different experiments conducted to validate the design strategy of CQS as well as its overall performance. We demonstrated that suggestions generated using CQS are often favored over those provided by Google, a search engine preferred by children. This is promising, given the fact that CQS does not depend on query logs, which are rarely, if at all, publicly available. Instead, CQS relies on public data sources, which are accessible through the Internet, to train probabilistic models that can be updated over time, allowing CQS to offer *timely* suggestions. Moreover, CQS generates suggestions on the fly, as opposed to the suggestions provided by many of the suggestion modules examined in our experiments, which are based on queries previously formulated by other users.

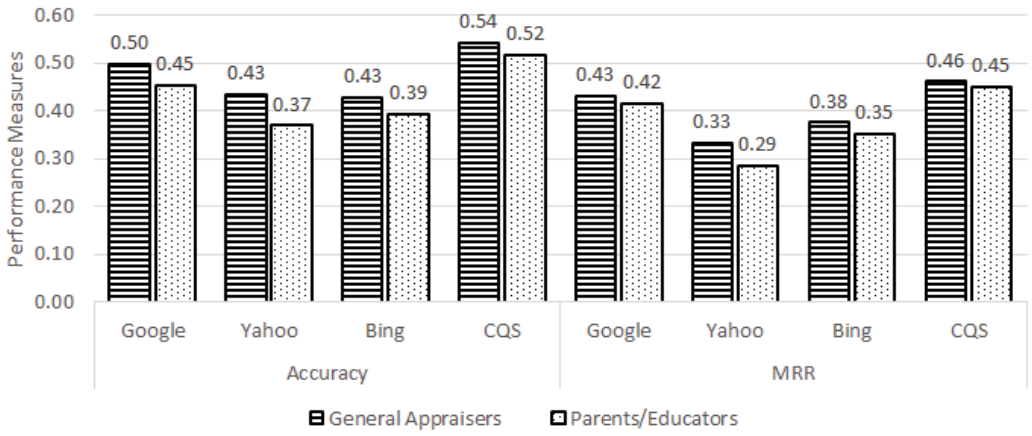

Fig. 10. Evaluations of CQS and other QS modules based on Mechanical Turk appraisers' responses collected during 2017

Furthermore, children are familiar and accustomed to the keywords specified in suggestions offered by CQS, since they are extracted from children's vocabulary and children web pages. Due to of its design methodology, CQS is tailored to serve groups of children of different ages.

5. Conclusions

A statistical report published in 2012 shows that 76% of children searched information on the Internet [16], which is a significant number back to those days and in today's standard. To enhance the children's web search experience, it is critical to design a query suggestion module that tailored towards children's information needs. In this paper, we have proposed a query suggestion module, called CQS, to suggest queries for children. Instead of following existing query suggestion approaches that rely on frequently-used queries in query logs or children's query suggestion approaches that count on snippets and titles given by search engines to (obtain tags that can be used to) generate candidate suggestions, CQS considers sentences in children's writing, children's vocabulary/phrases, simplicity of words, children's subject headings, and children's categories (i.e., subject areas) extracted from various children's websites, to generate simple and comprehensible phrases as query suggestions. The novelty of CQS is its reliance on freely and easily accessible online content/documents written by or for children. These resources not only allow CQS to make age-appropriate suggestions, but they also offer different children-likelihood features to be considered for capturing children's information needs, creating cohesiveness and simplicity of keywords in suggestions, and enriching the coverage of various topics in suggestions. Experiments conducted to evaluate the performance of CQS demonstrate the correctness of the design methodology of CQS and show that (i) children prefer suggestions offered by CQS over Yahoo! and Bing's, and (ii) the suggestions made by CQS are as popular as the ones recommended by Google.

References

[1] R. Benjamin. Reconstructing Readability: Recent Developments and Recommendations in the Analysis of Text Difficulty. *Educational Psychology Review*, 24(1):63–88, 2012.

[2] S. Bhatia, D. Majumdar, and P. Mitra. Query Suggestions in the Absence of Query Logs. In *Proceedings of ACM Conference on Research and Development in Information Retrieval (ACM SIGIR)*, pages 795–804, 2011.

[3] D. Bilal and R. Ellis. Evaluating Leading Web Search Engines on Children's Queries. In *Proceedings of International Conference on Human-Computer Interaction*, pages 549–558. Springer, 2011.

[4] D. Bilal and J. Kirby. Differences and Similarities in Information Seeking: Children and Adults as Web Users. *Information Processing & Management*, 38(5):649–670, September 2002.

[5] D. Bilal, S. Sarangthem, and I. Bachir. Toward a Model of Children's Information Seeking Behavior in Using Digital Libraries. In *Proceedings of International Symposium on Information Interaction in Context (IIiX)*, pages 145–151, 2008.

[6] G. Cormack, C. Clarke, and S. Buettcher. Reciprocal Rank Fusion Outperforms Condorcet and Individual Rank Learning Methods. In *Proceedings of ACM Conference on Research and Development in Information Retrieval (ACM SIGIR)*, pages 758–759, 2009.

[7] W. Croft, D. Metzler, and T. Strohman. *Search Engines: Information Retrieval in Practice*. Addison Wesley, 2010.

[8] A. Druin, E. Foss, L. Hatley, E. Golub, M. Guha, J. Fails, and H. Hutchinson. How Children Search the Internet with Keyword Interfaces. In *Proceedings of International Conference on Interaction Design and Children (ACM IDC)*, pages 89–96, 2009.

[9] C. Eickhoff, P. Dekker, and P. de Vries. Supporting Children's Web Search in School Environments. In *Proceedings of Information Interaction in Context Symposium (ACM IIIX)*, pages 129–137, 2012.

[10] C. Eickhoff, T. Polajnar, K. Gyllstrom, S. Torres, and R. Glassey. Web Search Query Assistance Functionality for Young Audiences. In *Proceedings of the European Conference on Information Retrieval (ECIR)*, pages 776–779, 2011.

[11] K. Flowers and N. Cookie. Knowledge Structure and Subject Access. In *Proceedings of Posters and Short Talks of the SIGCHI Conference on Human Factors in Computing Systems (ACM SIGCHI)*, page 3, 1992.

[12] T. Gossen. *Search Engines for Children: Search User Interfaces and Information-Seeking Behaviour*. Springer, 2016.

[13] K. Gyllstrom and M.-F. Moens. Wisdom of the Ages: Toward Delivering the Children's Web with the Link-based AgeRank Algorithm. In *Proceedings of ACM International Conference on Information and Knowledge Management (ACM CIKM)*, pages 159–168, 2010.

[14] B. Jansen, A. Spink, and T. Saracevic. Real Life, Real Users, and Real Needs: A Study and Analysis of User Queries on the Web. *Information Processing & Management (IPM)*, 36(2):207–227, 2000.

[15] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems*, 20(4):422–446, October 2002.

[16] Y. Kammerer and M. Bohnacker. Children's Web Search with Google: The Effectiveness of Natural Language Queries. In *Proceedings of International Conference on Interaction Design and Children (ACM IDC)*, pages 184–187, 2012.

[17] S. Karimi and M. Pera. Recommendations to Enhance Children Web Searches. In *Poster Proceedings of ACM Conference on Recommender Systems*, 2015. http://ceur-ws.org/Vol-1441/recsys2015_poster18.pdf.

[18] J. Lee. Analyses of Multiple Evidence Combination. In *Proceedings of ACM Conference on Research and Development in Information Retrieval (ACM SIGIR)*, pages 267–276, 1997.

[19] J. Lennon, P. Koleff, J. Greenwood, and K. Gaston. The Geographical Structure of British Bird Distributions: Diversity, Spatial Turnover and Scale. *Animal Ecology*, 70:966–979, 2001.

[20] Y. Li, A. Algarni, M. Albathan, Y. Shen, and M. Bijaksana. Relevance Feature Discovery for Text Mining. *IEEE Transactions on Knowledge and Data Engineering*, 27(6):1656–1669, 2015.

[21] T. Mitchell. *Machine Learning*. McGraw Hill, 1997.

[22] M. Motz, J Nachbar, M. Marsden, and R. Ambrosetti. *Eye on the Future: Popular Culture Scholarship into the Twenty-First Century in Honor of Ray B. Browne*. Popular Press, 1994.

[23] U. Ozertem, O. Chapelle, P. Donmez, and E. Velipasaoglu. Learning to Suggest: A Machine Learning Framework for Ranking Query Suggestions. In *Proceedings of ACM Conference on Research and Development in Information Retrieval (ACM SIGIR)*, pages 25–34, 2012.

[24] M. Pera and Y.-K. Ng. What to Read Next?: Making Personalized Book Recommendations for K-12 Users. In *Proceedings of ACM Conference on Recommender Systems (RecSys)*, pages 113–120, 2013.

[25] M. Shaikh, M. Pera, and Y.-K. Ng. A Probabilistic Query Suggestion Approach without Using Query Logs. In *Proceedings of IEEE International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 633–639, 2013.

[26] M. Shaikh, M. Pera, and Y.-K. Ng. Suggesting simple and comprehensive queries to elementary-grade children. In *Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, volume 1, pages 252–259. IEEE, 2015.

[27] S. Torres, D. Hiemstra, and P. Serdyukov. Query Log Analysis in the Context of Information Retrieval for Children. In *Proceedings of ACM Conference on Research and Development in Information Retrieval (ACM SIGIR)*, pages 847–848, 2010.

[28] S. Torres, D. Hiemstra, I. Weber, and P. Serdyukov. Query Recommendation for Children. In *Proceedings of ACM International Conference on Information and Knowledge Management (ACM CIKM)*, pages 2010–14, 2012.

[29] S. Torres, D. Hiemstra, I. Weber, and P. Serdyukov. Query Recommendation in the Information Domain of Children. *Journal of the American Society for Information Science and Technology (JASIST)*, 65(7):1368–1384, 2014.

[30] I. Vidinli and R. Ozcan. New Query Suggestion Framework and Algorithms: A Case Study for an Educational Search Engine. *Information Processing & Management*, 2016.

[31] N. Zhong, Y. Li, , and S.-T. Wu. Effective Pattern Discovery for Text Mining. *IEEE Transactions on Knowledge and Data Engineering*, 24(1):30–44, 2012.

Table 5
Children queries and their top suggestions provided by CQS and other web search engines

Children Query	**Top Suggestions Provided by Different Search Engines**			
	CQS	Google	Yahoo!	Bing
Art	Art and crafts	Art of manliness	Art van furniture	Art Institute of Chicago
Animal migration route	Animal migration route every time	Animal migration routes	Animal migration route of the monarch	Animal migration routes in washington state
Basketball player	Basketball player team	Basketball player with clothing line	Basketball player clipart	Basketball player with broken leg
Biggest	Biggest book fair	Biggest looser 2017	Biggest loser 2017	Biggest loser
Central train station	Central train station slackened pace	Central train station amsterdam	Central train station milan italy	Central train station amsterdam
Cheese	Cheese boxes	Cheese sauce	Cheese ball recipes	Cheese ball recipes
Cheese cake	Cheese cake sweet	Cheese cake factory	Cheese cake factory	Cheese cake recipes scratch
Chocolate chip cookie	Chocolate chip cookie dough	Chocolate chip cookie recipe	Chocolate chip cookie recipe	Chocolate chip cookie recipe
Cookie recipes	Cookie recipes in book	Cookie recipes without butter	Cookie recipes from scratch	Cookie recipes from scratch
Disney	Disney movie	Disney store	Disney character	Disney cruise
Dog	Dog pet	Dog breeds	Dog breeds	Dog breeds
Ice cream	Ice cream cake	Ice cream boise	Ice cream alley	Ice cream maker
India	India trading company	India news	India map	India news
Japan	Japan animation	Japan time	Japan airlines	Japan crate
Famous piano concert	Famous piano concert composers	Famous piano composers of all time	Famous piano composers of classical music	Famous piano composers list
Lego	Lego movie	Lego dimensions	Lego city	Lego games
Lego dinosours	Lego dinosaurs robots	Lego dinosaurs movie	Lego dinosaurs video	Lego dinosaurs games
Map	Map expert	Map of idaho	Yahoo maps	Mapquest
Meme	Meme in july	Meme maker	Meme generator	Meme generator
Minecraft	Minecraft	Minecraft skins	Minecraft skins	Minecraft servers
Music	Music books	Music notes	Music videos	Music notes
National football	National football coach	National football league	National football league	National football league
Migration routes	Migration routes enable people	Migration routes definition	Migration routes in the united states	Migration routes of birds
Most popular sport	Most popular sport music	Most popular sport in the world	Most popular sport in the world	Most popular sport in the world
Nike	Nike shoes	Nike shoes	Nike outlet	Nike inc.
Pet	Pet dog	Petsmart	Pet supplies	Pet supplies plus
Piano	Piano teacher	Piano guys	Piano guys	Piano guys
Popular book	Popular book contains municipality of the district	Popular book series	Popular book series	Popular book club
Soccer	Soccer ball	Soccer ball		Soccer games
Sport shoes	Sport shoes fashionist	Sport shoes for women	Sport shoes for women	Sport shoes for men
Star wars	Star wars characters	Star wars 8	Star wars rogue one	Star wars rebels
Star wars clones	Star wars clones episode	Star wars clones wars	Star wars clones vs droids	Star wars clones ebay
Tallest person	Tallest person of the year	Tallest person in the world	Tallest person in the world	Tallest person in the world
Tiger	Tiger Woods	Tiger Woods	Tiger Woods	Tiger Woods
Transformer	Transformers movies	Transformers toys	Transformers toys	Transformers calculator
Video	Video games	Video games	Video converter	Video editor
Water	Water bottles	Water cycle	Water softener systems	Water softeners
Youtube	Youtube celebrity	Youtube to mp3	Youtube music	Youtube music