TRAINING WHEELS FOR WEB SEARCH: MULTI-PERSPECTIVE LEARNING TO RANK TO SUPPORT CHILDREN'S INFORMATION SEEKING IN THE CLASSROOM

by

Garrett Allen

A thesis

submitted in partial fulfillment

of the requirements for the degree of

Master of Science in Computer Science

Boise State University

December 2021

BOISE STATE UNIVERSITY GRADUATE COLLEGE

DEFENSE COMMITTEE AND FINAL READING APPROVALS

of the thesis submitted by

Garrett Allen

Thesis Title: Training Wheels for Web Search: Multi-Perspective Learning to Rank to Support Children's Information Seeking in the Classroom

Date of Final Oral Examination: 18 December 2021

The following individuals read and discussed the thesis submitted by student Garrett Allen, and they evaluated the presentation and response to questions during the final oral examination. They found that the student passed the final oral examination.

Maria Soledad Pera, Ph.D.	Chair, Supervisory Committee
Jerry Alan Fails, Ph.D.	Member, Supervisory Committee
Casey Kennington, Ph.D.	Member, Supervisory Committee
Katherine Landau Wright, Ph.D.	Member, Supervisory Committee

The final reading approval of the thesis was granted by Maria Soledad Pera, Ph.D., Chair of the Supervisory Committee. The thesis was approved by the Graduate College.

ACKNOWLEDGMENTS

Throughout the writing of this thesis I have received a great deal of support and assistance.

First, I would like to thank my advisor, Dr. Sole Pera, whose incredible patience, knowledge, and belief in me consistently pushed me to the best of my abilities. Your impact on this work cannot be overstated, thank you.

I would also like to acknowledge Ashlee Milton, whose willingness to collaborate, discuss, and work through difficult hurdles ensured that this thesis came to a successful close. I greatly appreciate the sacrifice of your time towards my goals.

Finally, I would like to thank my parents, whose steadfast belief in me gave me the strength to not waver in my path. I love, and am forever grateful for, both of you.

ABSTRACT

Bicycle design has not changed for a long time, as they are well-crafted for those that possess the skills to ride, i.e., adults. Those learning to ride, however, often need additional support in the form of training wheels. Searching for information on the Web is much like riding a bicycle, where modern search engines (the bicycle) are optimized for general use and adult users, but lack the functionality to support non-traditional audiences and environments. In this thesis, we introduce a set of training wheels in the form of a learning to rank model as augmentation for standard search engines to support *classroom* search activities for *children* (ages 6–11). This new model extends the known listwise learning to rank framework through the balancing of risk and reward. Doing so enables the model to prioritize Web resources of high educational alignment, appropriateness, and adequate readability by analyzing the URLs, snippets, and page titles of Web resources retrieved by a given mainstream search engine. Experiments including an ablation study and comparisons with existing baselines showcase the correctness of the proposed model. Outcomes of this work demonstrate the value of considering multiple perspectives inherent to the classroom setting, e.g., educational alignment, readability, and objectionability, when applied to the design of algorithms that can better support children's information discovery.

TABLE OF CONTENTS

ABSTRACT v

LIST OF TABLES viii

LIST OF FIGURES ix

LIST OF ABBREVIATIONS x

1 Introduction 1

2 Related Work 7

2.1 Learning to Rank 7

2.2 Ranking Web Resources for Children 10

3 Methodology 13

3.1 Perspectives: From Theory to Practice 14

3.1.1 Readability 14

3.1.2 Educational Resources 16

3.1.3 Objectionable Resources 20

3.2 `REdORank`: From Theory to Practice 26

3.2.1 AdaRank 26

3.2.2 Multi-Perspective Optimization with Cost Sensitivity 29

4 Experimental Results 32

4.1 Finding a Readability Formula Fitting Web Resources 32

4.2 Detecting Educational Web Resources 39

4.2.1 Experimental Set-up 39

4.2.2 Results and Discussion 41

4.3 Identifying Objectionable Resources 44

4.3.1 Experimental Set-up 44

4.3.2 Results and Discussion 46

4.4 Re-ranking Web Resources with `REdORank` 48

4.4.1 Experimental Set-up 48

4.4.2 Results and Discussion 50

5 Conclusion 55

REFERENCES 59

A Data for Experiments 80

LIST OF TABLES

4.1 Performance analysis of BiGBERT (ablation study along with experiments related to comparisons with counterparts). The suffixes **-U** and **-S** indicate model applied to URL and snippet only, resp.; **-E** indicates model augmented with educational data. * and † significant w.r.t. BiGBERT and non-educational counterpart, resp. Significance determined with McNemar's test, $p<0.05$. 41

4.2 Evaluation of objectionable classification models using ObjSet. * indicates significance w.r.t. $Judge_{bad}$ determined by McNemar's test with Bonferroni Correction, $p<0.05$. 47

4.3 Performance of REdORank and ablation variations using RankSet. The suffixes -R, -E, -O indicate Readability only, Educational only, and Objectionable only, respectively. -M indicates the use of the mixer for educational alignment and readability, and -MER indicates the use of the mixer *with* -E and -R. * indicates significance w.r.t. REdORank and bold indicates best performing for each metric. 52

4.4 Performance of REdORank and baselines using RankSet. * indicates significance w.r.t.REdORank and bold indicates best performing for each metric. 53

LIST OF FIGURES

2.1 Framework for problem definition of learning to rank [52]. 8

3.1 The REdORank framework. REdORank re-ranks Web resources retrieved from a mainstream SE in response to a child's query formulated in a classroom setting by balancing reward with risk. 13

3.2 BiGBERT architecture (R_U and R_S denote the URL and snippet of a given resource R, resp.). 18

4.1 ER distribution across different grades for traditional readability formulas.36

4.2 Distribution of ER for Spache-inspired formulas across grades. 38

4.3 NDCG@10 for different re-ranking models using RankSet. 54

LIST OF ABBREVIATIONS

AoA – Age of Acquisition

BiGRU – Bidirectional Gated Recurrent Unit

CCSS – Common Core State Standards

CNN – Convolutional Neural Network

CS-DCG – Cost Sensitive Discounted Cumulative Gain

DCG – Discounted Cumulative Gain

ER – Error Rate

FNR – False Negative Rate

FPR – False Positive Rate

GCS – Google Custom Search

ICS – Idaho Content Standards

IDLA – Idaho Digital Learning Alliance

IR – Information Retrieval

LTR – Learning to Rank

MLM – Masked Language Model

MP-LTR – Multi-perspective Learning to Rank

MRR – Mean Reciprocal Rank

NDCG – Normalize Discounted Cumulative Gain

NGSS – Next Generation Science Standards

ODP – Open Directory Project

RAZ – Reading A-Z

SE – Search Engine

SERP – Search Engine Result Pages

URL – Uniform Resource Locator

CHAPTER 1

INTRODUCTION

Children in elementary classrooms (Kindergarten–4^{th} grade) often use search engines (**SE**) to find Web resources needed to complete their school assignments [13, 105]. Among SE built specifically for children's use in a classroom environment, we find current solutions require regular maintenance, such as EdSearch[1] and Kidtopia[2]. EdSearch relies on manual curation of resources (e.g., text or media) to identify educational ones. Kidtopia instead offers resources from a selection of white-listed sites using Google's Custom Search (**GCS**) platform, which utilizes the *SafeSearch* feature to filter out pornographic resources. The white-listing via manual curation restricts the sites to be both age-appropriate and educational, but as the Web grows at a rapid rate, maintaining an up-to-date white-list becomes burdensome. Moreover, children's SE based on GCS are known to return less relevant results nearly 30% of the time, trading relevance for safer results [42]. In addition to these inefficacies, specialized SE must also overcome the barrier of adoption: children prefer to use the popular mainstream options for SE, such as Google or Bing [32].

Mainstream SE are designed and optimized for adults, and therefore can overlook unique factors that impact children using them. For instance, children face many barriers related to query formulation, some of which researchers have attempted

[1] https://www.lumoslearning.com/llwp/edsearch.html

[2] https://www.kidtopia.info/

to offer aid for [16, 82, 123]. In addition to well-studied barriers related to query formulation, children also struggle to recognize what and how much information is available online, seldom looking past the first six resources presented on a search engine result page (**SERP**) [43]. Children also have trouble understanding the content of retrieved resources due to the complexity of their texts, which leads to uncertainty with relevant resource selection [7]. When turning to mainstream SE, children can also be inadvertently exposed to resources that are inappropriate for their consumption. This is an unfortunate side effect of functionality, like Google's *SafeSearch*, offered by mainstream SE primarily filtering pornography [131] and not accounting for other potentially harmful content, e.g., violence. Safe search functionality also suffers from over-filtering by preventing resources from being returned if they contain terms that might be mistaken as inappropriate [12].

We aim to advance knowledge in the area of Information Retrieval (**IR**) for children, and more specifically better enable children's access to online information via SE. As they grow, children require different levels of support from the SE they interact with. As a starting point in our exploration, we focus on tailoring SERP for specific audiences and contexts. To define the scope for our work, we turn to the framework from [70] that allows for the comprehensive design and assessment of search systems for children through four pillars. Defined for our work, these pillars are: children aged 6–11 in grades Kindergarten–4 (**K–4**) as the *user group*, classrooms as the *environment*, information discovery as the *task*, and re-ranking of resources to fit audience and context as the *strategy*. Guided by the pillars, we introduce `REdORank`, a novel re-ranking framework based on multi-perspective learning to rank (**LTR**) meant to support *children*'s use of their preferred SE to complete *classroom*-related tasks. `REdORank` builds on the demonstrated retrieval effectiveness of mainstream SE, as well as their ability to respond to any given query due to their large supply of indexed pages [75]. Given a child's search query, REdORank examines and re-ranks resources retrieved by mainstream SE in a manner that those which align with educational standards and are readable by children are ranked higher, whilst those that contain material inappropriate for viewing by children in classrooms are pushed lower in the result list.

For REdORank to learn how to prioritize resources to best suit our user group and environment, it examines three distinct perspectives: **readability**, i.e., "the overall effect of language usage and composition on readers' ability to easily and quickly comprehend the document" [85], **educational alignment**, and **objectionability**. Educationally aligned resources are defined as those that align with the guidelines presented in the Common Core State Standards (**CCSS**). These guidelines provide a set of learning outcomes for each grade K–12 that students are expected to achieve. For example, a grade 1 learning outcome from the CCSS states "Identify the main topic and retell key details of a text" [63]. On the other hand, objectionable resources are resources that contain content beyond pornography that is inappropriate for children in a classroom. Both educational alignment and readability act as "reward" perspectives, i.e., signals that should be optimized to increase the rank of a resource, whereas objectionability serves as a "risk" perspective. A ranking strategy known as multi-perspective LTR is employed where a ranking model learns a ranking function that prioritizes resources for more than one perspective. By prioritizing resources that align with the readability levels of our user group, REdORank benefits our target audience as children that read over their reading level experience lower reading comprehension [8].

Estimating grade levels of online resources is not a simple matter, given the broad range of formulas available for readability or grade level estimation. In addition, there exists no consensus on which of the available formulas should be used for online resources. Consequently, we examine formulas and leverage in `REdORank` the one most effective for predicting the readability levels of resources targeting the reading abilities of stereotypical 6–11 year olds.

Responding to our environment, `REdORank` considers the educational alignment of resources and aims to promote those with educational value as previous research has shown that ranking educational resources higher in search results has the potential to increase learning efficiency [117]. `REdORank` determines educational alignment through analyzing the URL and snippet of resources using text representation strategies such as domain-specific embeddings and BERT [11, 33].

As previously stated, not everything on the Web is appropriate for children, which brings us to the objectionability perspective. Preventing the display of inappropriate results while also avoiding over-filtering results that may appear as objectionable but are not, e.g., an article on breast cancer [42], requires a solution that goes beyond safe search. Therefore, we go beyond pornography and consider other sources of objectionability, such as violence, drugs, or guns. `REdORank` utilizes an approach that applies a cost to a resource's ranking based on a determined likelihood that a resource is objectionable. By learning to simultaneously maximize resources with educational alignment and readability, while minimizing those with objectionability, `REdORank` is well suited to support children's search activities in the classroom.

We posit that a *LTR* strategy can be augmented to simultaneously consider multiple traits of online resources in order to yield a SERP that prioritize *educationally valuable* and *comprehensible* resources while minimizing those that are *objectionable.* To guide the work pertaining to these topics, this thesis addresses the following research questions:

1. Which readability formula simultaneously suits resource type, context needs, and user group outlined for our task?

2. Do snippets along with URLs help identify educational resources? Does domain-specific knowledge affect identification of educational resources?

3. Can topic-specific lexicons empower the identification of objectionable Web resources?

4. Does the adaptation of an LTR model to account for multiple perspectives lead to the prioritization of resources that are relevant to both children and the classroom setting?

The primary contribution of this work is an LTR framework optimized on multiple perspectives simultaneously for children's search, which to the best of our knowledge is the first such multi-perspective LTR framework. Further contributions include a model to determine educational alignment of online resources, the identification of a readability formula that is effective for calculating the reading level of online resources, and a model that identifies objectionable resources beyond the limited scope of safe search. Our work can help facilitate how children access educational content online and thus supports classroom instruction. In fact, `REdORank` can be used in conjunction with any SE, and when so combined can provide support for search as learning among K–4^{th} grade students [59, 120, 129]. The educational alignment model has the potential to support teachers identifying online resources to leverage in the classroom [39]. Finally, identifying a formula that accurately estimates the reading level of online resources could inform future design of recommender systems or other online systems tailored to young readers [74].

The rest of this manuscript is organized as follows. In Chapter 2, we offer background information pertaining to LTR for Web search and discuss ranking strategies that aim to support children's use of SE. Thereafter, in Chapter 3, we detail the design of `REdORank`; this is followed by the in-depth empirical analysis presented in Chapter 4, which we conducted to verify the performance of our re-ranking framework and to assess the need to include all perspectives in the ranking. Lastly, in Chapter 5, we present some concluding remarks, limitations, and future research directions inspired by the work presented in this thesis.

CHAPTER 2

RELATED WORK

In this chapter, we provide background information on LTR and discuss existing ranking strategies that tailor the (retrieval and) ranking of resources for children.

2.1 Learning to Rank

LTR is a machine learning strategy that, when applied to Information Retrieval, creates a task in which the goal is to automatically determine a ranking model using training data, such that the model constructed can sort new resources using a learned ranking function according to resources' degrees of relevance, preference, or importance [79]. LTR can be expressed in terms of queries, labels, and resources. Given a set of m queries $Q = \{q_1, ..., q_m\}$, there exists a set of k resources $R_m = \{r_{m,1}, ..., r_{m,k}\}$ for query in q_m. Similarly, there exists a set of labels $y_m = \{y_{m,1}, ..., y_{m,k}\}$ for each resource R_m. Let $f(q_m, r_{m,k})$ be a ranking function that calculates a ranking score for a query-document pair, and let $\ell(f; q_m, r_{m,k}, y_{m,k})$ be a loss function for the prediction of the function f over the query-document pair $(q_m, r_{m,k})$. The LTR problem can then be defined as seeking to find the optimal ranking function f_{opt} (in Equation 2.1) through the minimization of the loss function over a labelled training set [76, 79, 52]. We depict the framework for this problem definition in Figure 2.1.

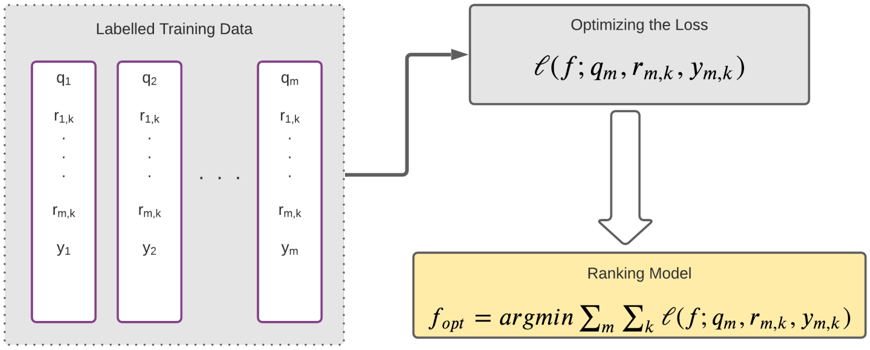


Figure 2.1: Framework for problem definition of learning to rank [52].

$$f_{opt} = argmin \sum_m \sum_k \ell(f; q_m, r_{m,k}, y_{m,k}) \tag{2.1}$$

Over time, advancements in LTR models have expanded on the loss function to accept more than one resource as input. As a result, the following categorizations for LTR models have arisen: pointwise, pairwise, or listwise [76], based on whether a single resource, a pair of resources, or a list of resources, respectively, are operated over during optimization of the loss function.

Regardless of the category they belong to, LTR models have been successfully applied to various areas of IR, such as question answering [34], document retrieval [79], recommendation [77, 96, 128], and, most prominently, Web search [81, 95]. When used for Web search, models using listwise loss functions have been shown to be more effective in terms of ranking accuracy and degree of certainty of ranking accuracy in relation to the pointwise and pairwise counterparts [20, 118]. There exists a number of well-known listwise-based models, including *AdaRank* [127], *ListNet* [20], *ListMLE* [125], *online-listMLE* [80], *SetRank* [97], and *U-Rank* [30]. Each of these models present a step towards advancing knowledge pertaining to LTR, yet all optimize their respective ranking functions on a single relevance measure.

In practice, the degree of relevance of a search result is not always established based on a single trait. For instance, a user searching for a seafood restaurant for dinner would consider location, price, and reputation as factors informing relevance. Students searching for information on John Adams for a class assignment would instead determine resource relevance by considering factors such as whether a resource uses language they can understand, whether the John Adams being discussed is the correct individual, and whether the resource discusses the aspect of John Adams for which they are seeking information, i.e., information on his term as President vs. information on his role during the American Revolution. To better align with such real world scenarios, multi-objective LTR strategies that optimize loss functions for multiple measures of relevance have been brought forth [18, 116, 122]. Carmel et al. [21] use label aggregation to reduce a multi-objective problem to a single objective one, followed by applying LambdaMART [19] to optimize for the aggregated labels. Momma et al. [90] also make use of LambdaMART, combining Augmented Lagrangian, a process of introducing an explicit Lagrange multiplier into the loss function being optimized [93], to create a model that handles constrained optimization by “iteratively solving unconstrained problems.” While these strategies expand LTR from single objective to multi-objective, both still opt for a pairwise approach. When accounting for multiple objectives, listwise approaches like AdaRank are rarely considered. Given AdaRank’s applicability for Web tasks [20, 118], this is the LTR variation we incorporate as part of `REdORank`’s design.

2.2 Ranking Web Resources for Children

When seeking for information using mainstream SE, children tend to (i) explore SERP produced in response to their queries using a sequential process from top to bottom, and (ii) click higher-ranked results [35, 43, 50, 53]. As such, it is imperative for these mainstream SE to prioritize retrieved resources relevant to the information needs of children. Existing attempts to address this requirement include the work by Miltsakaki [88], who sorts resources with respect to a user-defined reading level (for middle and high school students) and the resource's readability as calculated using the Coleman-Liau Index [25] together with the LIX and RIX formulas [9]. Similarly, Collins-Thompson et al. [26] re-rank results matching user reading levels inferred from their search history.

Beyond readability, Gyllstrom and Moens [54] introduce *AgeRank*, a modified version of *PageRank* that leverages websites for younger audiences, following the premise that sites designed for children are more likely to link to other child-friendly sites. Syed and Collins-Thompson [117] present a search algorithm that re-ranks results for learning utility through an analysis of keyword density, assuming that a user exposed to more keywords in fewer resources will learn information on a particular subject more successfully. The aforementioned strategies prioritize resources using only a single perspective, yet when serving a particular user group in a specific context, considering only one perspective restricts the results that can be retrieved. We hypothesize that by incorporating more perspectives, such as educational alignment and readability, a more varied set of results can be provided that better serve the user group and context.

Research pertaining to education-based ranking is rich, resulting in strategies based on topic modelling, term clustering, quality indicators, and collaborative filtering [101, 109, 111, 100]. Notable examples include the work by Marani [83], i.e., *WebEduRank*, who defines a teaching context (a representation of the requirements and experiences of an instructor), which is used to rank learning objects to support instructors. Estivill-Castro and Marani [41] also rank resources for instructors by analyzing the suitability of a resource for teaching a concept. Acuña-Soto et al. [2] consider students as part of their audience in their work to rank math videos using a multi-criteria decision making framework. Unfortunately, as with readability and child-friendliness, some of these works do not target children as the intended user group, and the majority focus on a single perspective.

Focusing on children in an educational context, Usta et al. [121] train an LTR model for a query-dependent ranking strategy aimed at prioritizing educational resources for students in the 4^{th}–8^{th} grades. Through feature engineering, the authors extract disjoint sets of features from the query logs of a Turkish educational platform called Vitamin [120]: (i) query-document text similarity, (ii) query specific, (iii) document specific, (iv) session based, and (v) query document click based. Unique to this approach is that within the query specific and document specific groups are domain-specific features such as the course, grade, and document type, e.g., lecture, video, or text. This approach differs from ours in that the features used in training a ranker uses data originating from a domain-specific SE that includes course and grade information of the resources whereas we design a re-ranker that is SE agnostic, allowing our re-ranker to be coupled with any generic SE. Additionally, the features used by Usta et al. [121] include click data originating from children, which is not readily or publicly available for our user group.

The strategy most closely related to `REdORank` is *Korsce* [87]. This multi-perspective strategy examines appropriateness, curriculum alignment, objectivity, and reading comprehensibility of resources to identify those that best match 3^{rd} – 5^{th} grade children searching in the classroom. Korsce treats resources as inappropriate if they refer to pornography and hate-speech, but fails to account for other potentially objectionable topics like alcohol or drugs. For curriculum alignment, the authors adopt a topic modelling approach with Latent Dirichlet Allocation (LDA) as a way to estimate the degree in which a resource is related to curriculum. This approach follows a word-level and semantic space exploration of resources, but does not take into account the contextual information that can be garnered from considering resource text in its entirety.

When considering reading comprehension, Milton et al. [87] introduce a formula that estimates a score based on the Flesch-Kincaid readability formula and a cosine curve that penalizes resources whose readability level is beyond the expected grade level of a user. There are two major gaps in this approach: (i) the selection of the Flesch-Kincaid formula is based on popular use rather than empirical exploration, which we conduct in our work, and (ii) the reading comprehension score requires the knowledge of an expected grade for the user, which we cannot assume to know for our user group as we are focusing on multiple different grades.

Furthermore, Korsce ranks resources according to a static set of optimal weights [122]. These weights are manually chosen as the result of an empirical exploration of near-optimal rankers, where the optimal ranker is determined qualitatively. The selected ranker generates scores on a resource by resource basis (akin to pointwise LTR methods), leading to relative rankings based on the calculated scores. Alternatively, we utilize a listwise approach, allowing for absolute relevance comparisons between the resources as all resources are considered at once instead of independently.

CHAPTER 3

METHODOLOGY

In this chapter, we describe REdORank, a multi-perspective learning to rank (**MP-LTR**) framework that re-ranks resources through examining *in tandem* the Readability, Educational alignment, and Objectionability of each resource R retrieved by a mainstream SE in response to a child's query inquiring on classroom-related concepts. Taking advantage of the retrieval power of mainstream SE and directly informed by the aforementioned perspectives, REdORank identifies and prioritizes resources intended for K–4 classrooms and students. As shown in Figure 3.1, REdORank consists of three modules: the reward module, the risk module, and a balance module. Each module serves a specific purpose in the overall framework.

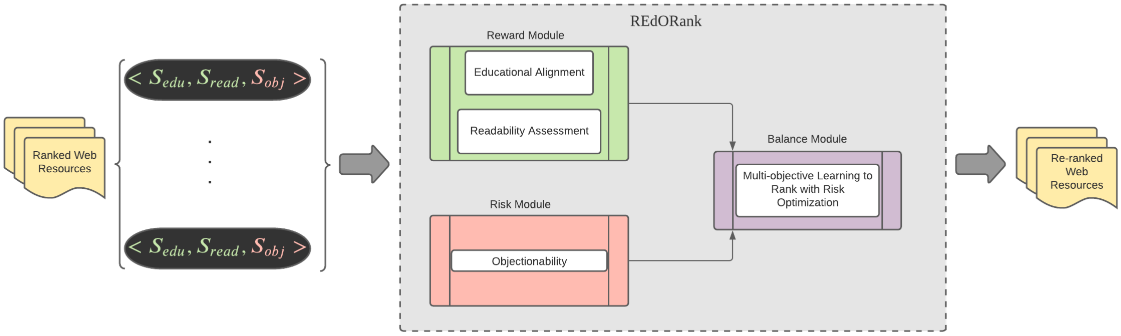


Figure 3.1: The REdORank framework. REdORank re-ranks Web resources retrieved from a mainstream SE in response to a child's query formulated in a classroom setting by balancing reward with risk.

The reward module determines the interaction between "positive" perspectives for resource analysis: readability and alignment with classroom curriculum. The risk module looks at the interaction of "negative" perspectives that identify resources as inappropriate for the user group. The balance module trades-off outputs of the risk module (a value that acts as cost and therefore decreases resource prioritization) and the reward module (a value meant to increase resource prioritization in the ranking) resulting in a final ranking score by which resources are reordered.

3.1 Perspectives: From Theory to Practice

REdORank re-ranks online Web resources by reassessing them according to three perspectives connected to children's information seeking in the classroom. To properly re-rank resources, we must quantify each perspective. In the remainder of this section, we describe how we represent the educational alignment, readability, and objectionability perspectives.

3.1.1 Readability

Readability is an important factor in supporting children's Web search as text complexity and therefore comprehension influences the degree to which a resource is relevant to a user [8, 107]. This makes it imperative to take into account the text complexity of retrieved resources when it comes to determining their position in a SERP. This is a nontrivial problem. There exists a plethora of formulas for text complexity estimation, from traditional ones based on shallow features, e.g., average words per sentence, to more complex ones based on deep learning [14, 36, 45, 89]. Even so, there is still a lack of consensus as to which formula to use for the automatic estimation of text complexity for Web texts.

As the overall performance of `REdORank` is directly impacted by the choice of readability formula, we evaluate several of them[1] and juxtapose their aptitude for readability assessment through assigning of grade levels based on their applicability to our user group, to our resource type, and the ease of calculation. As a result, the best suited formula in our analysis is `Spache-Allen`, which improves upon the original Spache [113] formula by expanding the vocabulary used to determine "easy words". The Spache formula relies on a vocabulary for easy words comprised of 1,064 words. This vocabulary, however, only contains terms deemed easy for children to comprehend, gathered in the 1970's from news and magazine articles for adults [36]. Instead, `Spache-Allen` accounts for terminology children are exposed to online by including 47,712 terms extracted from children's websites; a list of terms originally compiled in [82]. Moreover, `Spache-Allen` takes into consideration terminology that children learn through instruction. For this, we turn to the 30,000 words in the Age of Acquisition (**AoA**) dataset [68]. In the end, `Spache-Allen`, shown formally as Equation 3.1, relies on a vocabulary to identify easy words that includes 65,669 unique terms.

$$\textit{Spache-Allen}(T) = (0.141 \times w_T/s_T) + (0.086 * dif(T)) + 0.839 \tag{3.1}$$

where T is a text, and w_T and s_T are the number of words and sentences in T, respectively. The function $dif(T)$ determines the percentage of difficult words in T, where a word is deemed difficult if it does not appear in the augmented version of easy words vocabulary.[2]

[1] We provide details of the empirical exploration and vocabulary expansion with respect to that of the original Spache formula [113] in Section 4.1.

[2] `https://github.com/Neelik/spache-allen-vocabulary`

REdORank determines the readability score S_{read} of R (Equation 3.2) based on the estimation of readability determined using Spache-Allen, as applied to the corresponding snippet R_S.

$$S_{read}(R) = \textit{Spache-Allen}(R_S) \tag{3.2}$$

3.1.2 Educational Resources

Resource prioritization by readability can help support children's Web search. Yet, on its own, this perspective can overlook the environment that is the focus of our study, i.e., not all resources aligned with the reading abilities of children are suitable for the classroom. Thus, in the design of REdORank we also consider the educational value of resources, which we determine through a Web classification process.

Web resource classification is a well-explored area in Information Retrieval [56]. Recently, the field has seen an influx of research related to domain-specific classification, especially within the legal, financial, and medical domains [40, 60, 132]. Classification in the domain of education, however, remains relatively unexplored. As a broad term, education applies to a variety of classification tasks. Prior work includes classifying educational resources based on "the strength of the educative resource [as] a property evaluated cumulatively by the target audience of the resource (e.g., students or educational experts)" using a Support Vector Machine (SVM) [57]. This model, however, relies heavily on manually-annotated data and is applicable only to computer science education. Xia [126] also uses an SVM to classify resources supporting instruction, whereas EduBERT [24] detects college-level forum posts written by struggling students. In general, efforts in this area classify resources for unspecified age groups, adult students, limited subject areas, instructors, or institutional-level insights. There is a gap in the literature regarding recognizing educational Web resources for children ages 6–11 in grades K–4.

Regardless of the domain, classifiers tend to rely on features inferred from HTML page content [38, 114]. Processing full Web pages requires high computational power, large data storage, and time to retrieve [104] as Web pages are often dynamic and contain pictures, videos, or scripts in addition to text [105]. To address some of these constraints, state-of-the-art approaches examine only URLs [49, 105]. Unfortunately, URLs are not always comprised of meaningful tokens (i.e., valid terms), which may cause misclassifications. Consider the URL *https://www.youtube.com/watch?v=pX3V9hoX1eM* for a YouTube video by National Geographic For Kids related to animals. In this case, meaningful tokens include "youtube" and "watch," neither of which indicates the child-friendliness of the corresponding resource.

Mindful of the aforementioned limitations, we rely on domain knowledge obtained from Educational standards along with URL and descriptive text to inform the recognition of children's educational Web resources. Educational standards, such as the United States' CCSS and the Next Generation Science Standards (**NGCS**), provide learning outcomes for K–4^{th} grade students. In particular, we focus on educational resources that inform on subjects for grades K–4^{th}, such as language arts, science, and social studies, described in CCSS, NGCS, and the Idaho Content Standards (**ICS**).

As illustrated in Figure 3.2, BiGBERT, the Bidirectional Gated Recurrent Unit (**BiGRU**) with BERT model we introduce to recognize educational Web resources for children, has two main components: a URL and a snippet vectorizer. Given R, BiGBERT first vectorizes its URL, combining the domain-specific embeddings from *Edu2Vec* [11] with a BiGRU and a self attention layer. Shen et al. [112] show

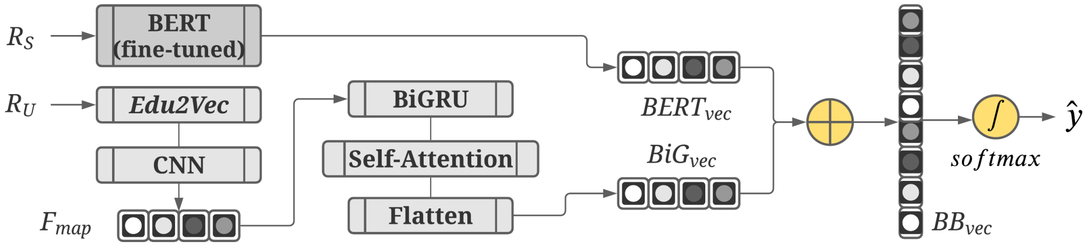


Figure 3.2: BiGBERT architecture (R_U and R_S denote the URL and snippet of a given resource R, resp.).

that using summaries instead of full page content results in comparable classification performance, thus we use snippets in place of full content. To vectorize R's snippet, BiGBERT fine tunes the transformer model BERT [33] using educational standards. Lastly, BiGBERT concatenates the snippet and URL vectors and applies a softmax function to determine the class of R.

URL Vectorizer. BiGBERT tokenizes R's URL (R_U) into a sequence of terms T by splitting on non-alphanumeric symbols (e.g., periods, dashes and forward slashes) and using SymSpell [47] to perform word segmentation as URLs tend to compound words together (e.g., changing *stackoverflow* to *stack overflow*). Each token $t_i \in T$ is mapped to its corresponding word embedding. If t_i is not part of the embedding dictionary, we attribute this to a possible misspelling or spelling variation, and thus attempt a correction using a single edit distance operation (i.e., replacing, adding, or removing a character). If t_i is still not in the dictionary, we discard it to ensure only meaningful tokens remain.

To learn a representation of R_U, BiGBERT uses the *Edu2Vec* word embeddings dictionary [11] as it incorporates domain knowledge from NGCS, CCSS, and ICS. These standards serve as structured knowledge sources to identify terms, topics, and subjects for K-4 grades, enabling BiGBERT to emphasize K-4 curriculum concepts in

R_U that may be overlooked by general-purpose pre-trained embeddings. Rather than analyzing independent embeddings, we design BiGBERT to scrutinize context-sensitive indicators from T. Inspired by Rajalakshmi and Aravindan [103] and in response to URLs not following traditional language syntax, we examine groups of embeddings (i.e., trigrams) using a Convolutional Neural Network (**CNN**)–a fast, effective, and compact method [65] to generate feature vectors from trigrams. The convolution results in a feature map F_{map}=<F_1,F_2,...,F_x>, $\forall_{f=1..x}$ F_f=$relu(w.x_{i:i+m-1}+b_u)$, where the rectified linear function $relu$ is applied to the dot product of a kernel w with a window of embeddings $x_{i:i+m-1}$ in T of size m=3; b_u is a bias term. To explore long distance dependencies of features that may appear far apart BiGBERT uses a BiGRU network, as it captures context information in a forwards and backwards direction. A self-attention layer then determines the importance of features identified by the CNN and BiGRU. This is followed by a flatten and dense layer that yields a single feature vector representation of R_U of size 128, denoted as $\boldsymbol{BiG_{vec}}$.

Snippet Encoding. As snippets are a few sentences long, unlike URLs which are at most a few words, we require a model that can scrutinize each snippet (R_S) as a whole. Hence, we incorporate the state-of-the-art transformer model BERT [33] into BiGBERT's design. BERT's ability to process sequences up to a maximum size of 512 tokens enables BiGBERT to exploit the sequential, contextual information within R_S in its entirety. Additionally, BERT's architecture consisting of 12 transformer blocks and self-attention heads ensures the learning of rich contextual information from each snippet. As such, we tokenize R_S into a sequence of sentences, encode it to BERT's specifications, and use BERT to attain an aggregate feature vector representation of size 768, denoted as $\boldsymbol{BERT_{vec}}$.

On domain-dependent tasks like the one we address here, BERT benefits from fine-tuning [115]. Thus, we adjust the traditional BERT to our definition of education by exploiting established educational standards. We perform fine-tuning as described in [115], training[3] BERT embeddings as an educational text classifier by adding a linear classification layer which uses binary cross entropy as loss and the Adam optimizer with learning rate=$1e^{-5}$.

Predicting Educational Alignment. To leverage evidence of educational alignment inferred from R_U and R_S, we concatenate BiG_{vec} with $BERT_{vec}$ as BB_{vec}. Using a fully connected layer on BB_{vec} with a softmax activation function, `BiGBERT`[4] produces a probability distribution $\hat{\mathbf{y}}$ over each class, educational and not, such that $\hat{y} \in [0, 1]$.

The softmax function ensures that the sum of the probabilities per class is one. Using `BiGBERT`, we define an educational alignment score S_{edu} for R (Equation 3.3). As one of two "reward" perspectives, `REdORank` utilizes this score in the determination of R's relevance gain.

$$S_{edu}(R) = BiGBERT(R_S, R_U) \tag{3.3}$$

3.1.3 Objectionable Resources

The Web contains an ever-growing collection of resources for users of many ages, experience, and knowledge levels. It is therefore anticipated for some of these resources to be more attuned with some user groups than others. Given the user group and environment that are the focus of this work, it becomes imperative for `REdORank`

[3] For fine-tuning we use 2,655 text passages from NGCS, CCSS, and ICS along with 2,725 from the Brown corpus [17, 44].

[4] `BiGBERT` is trained using a batch size of 128, binary cross-entropy loss function, and RMSProp optimizer [119] with momentum=0.2 and learning rate=0.001.

to mitigate the risk of presenting resources towards the top of SERP that could be deemed inappropriate. This is why REdORank incorporates an objectionability perspective to its design.

Establishing what makes a Web resource objectionable for children in a classroom setting is not trivial. Specialized SE built on the GCS platform (e.g., Kiddle and KidRex) use the safe search feature to eradicate objectionable resources, but still display promoted resources at the top of SERP in the form of advertisements. These adverts can redirect children to different SERP without the safe search protection, inadvertently exposing them to inappropriate material [42]. Safe search functionality is not without fault as unsuitable resources can make it past its filters [37, 12]. For example, in response to the query “dog facts”, Google with SafeSearch enabled retrieves a brewery website[5]. At the same time, there are websites that may appear objectionable but are not, e.g., an article on breast cancer [42]. Preventing the display of such results while also avoiding over-filtering is a difficult problem requiring a solution that goes beyond safe search.

Patel and Singh [98] augment GCS filtering capability by also considering resources containing hate speech or violence. Lee et al. [73] go even further, and filter content referring to abortion, alcohol, tobacco, illegal affairs, drugs, gambling, marijuana, pornography, violence, racism, and weapons. Unfortunately, their proposed filtering strategies rely on click-through data, which seldom exists for our user group. Milton et al. [87] introduce a click agnostic strategy to identify resources that are inappropriate for the classroom. The strategy leverages lexicons to account for the presence of sexually explicit and hate speech terms, in addition to misspelled terms, in the content, meta-tags, and anchor-tags of resources. This strategy, however,

[5] https://www.flyingdog.com/

treats as objectionable resources concerning pornography and hate speech, only two of the several topics that can be considered inappropriate for the user group and environment under study. Additionally, the full content of a resource is considered, which can be time consuming to process when deployed in a live scenario.

Observant of the strengths and limitations of existing filtering strategies, we adopt a simple, yet effective, technique for identifying resources deemed objectionable: $Judge_{bad}$. Given R, we first create a representation that captures its terminology from various categories. Treating the identification of R as objectionable as a *binary classification task*, we then employ a Random Forest model, which have been shown to maintain effectiveness even when compared to recent neural solutions [28, 29].

Objectionable categories. To account for the large variety of objectionable material present online, and inspired by prior strategies to detect objectionable resources [87, 73], we treat as objectionable for children in the classroom resources that relate to any category in ObjCat: Abortion, Drugs, Hate Speech, Illegal Affairs, Gambling, Pornography, and Violence. Note that the Drugs category refers to resources over-arching *drugs*, but also *alcohol*, *tobacco*, and *marijuana*. Further, Violence focuses on violent content, as well as *weapons*; Hate Speech accounts for *racism* and hateful/offensive content.

As previously stated, for determining the likelihood of resources being objectionable, we adopt a technique that scrutinizes their terminology and therefore requires the existence of pre-defined lists of 'objectionable' terms. In the case of Pornography and Hate Speech categories, we use the pre-defined lists used in [87], which are sourced from Google's archive[6] and the Hate Speech Movement's website[7], respec-

[6] `https://code.google.com/archive/p/badwordslist`

[7] `HateSpeechMovement.org`

tively. Unfortunately, there are no curated term lists associated with the remaining categories in ObjCat. Thus, we generate them through a novel process called category understanding via label name replacement [86].

We use websites from Alexa Top Sites [6] known to belong to categories appearing in ObjCat as our corpus for generating the term lists. For each category, excluding Pornography and Hate Speech, the occurrence of the category name (as well as subcategory names, if available) within a website from the corpus is masked and a pre-trained BERT encoder is used to produce a contextualized vector representation h with the masked category name. BERT's masked language model (**MLM**) head produces a probability distribution that a term w from within BERT's vocabulary will occur at the location of the masked category name.

Terms can occur in different contexts within the same corpus. Thus, terms in the extracted vocabulary are ranked by their probability of occurrence (Equation 3.4), and by how many times each term can replace a category name in the corpus while maintaining context.

$$p(w \mid h) = Softmax\left(W_2\, \sigma\left(W_1 h + b\right)\right) \tag{3.4}$$

where $\sigma(\cdot)$ is the activation function; W_1, W_2, and b are learned parameters for the masked language prediction task, pre-trained within BERT.

As in [86], we select the top 100 terms per category (or the entire list if less than 100 are extracted) as the final representative term list that captures contextually similar and synonymous terms associated with the corresponding categories.

Snippet representation. Due to the complexities of gathering, the computing resource, and storage needs for processing the full content of Web pages (Section

3.1.2), we use snippets as a proxy for the full page content.

We represent R with a collection of 16 text-based features extracted from its snippet. Seven of these features account for the prevalence (i.e., frequency of occurrence) of objectionable terms in R_S. For each category oc in ObjCat, we calculate the term prevalence, i.e., $TP(R_S, oc)$, as in Equation 3.5.

$$TP(R_S, oc) = \frac{\sum_{t \in TL_{oc}} tf(t, R_S)}{|R_S|} \tag{3.5}$$

where TL_{oc} is the term list for oc, t is a term in TL_{oc}, and $tf(\cdot)$ is a function that calculates the number of times t appears in R_S. Serving as a normalization factor, $|R_S|$ is the length of R_S after tokenization, punctuation & stop word removal, and lemmatization (using the NLTK[8] Python library).

We also consider the coverage of objectionable terminology in R_S, using seven features that account for scenarios where a term could be misconstrued as objectionable depending on context. For example, "breast" could occur frequently in a biology resource that is itself not objectionable; it can also appear in a pornographic resource. For each category oc, we calculate objectionable term coverage in R_S, i.e., $TCov(R_S, oc)$, using Equation 3.6.

$$TCov(R_S, oc) = \frac{\sum_{t \in TL_{oc}} \delta\left(t, R_S\right)}{\mid TL_{oc} \mid} \tag{3.6}$$

where TL_{oc} and t are as defined in Equation 3.5, $\delta(t, R_S)$ is 1 if t occurs at least once in R_S and 0 otherwise, and $|TL_{oc}|$, which is the total number of terms in TL_{oc}, acts as a normalization factor.

We explicitly account for misspelled terms, as producers of objectionable online

[8] `https://www.nltk.org/`

content are known to introduce intended misspellings as an attempt to bypass safe search filters [87]. We look at the prevalence of misspelled terms in R_s–how often misspellings occur in R_s–using Equation 3.7.

$$MP(R_s) = \frac{\sum_{t \in R_S} \beta\left(t, R_s\right)}{|R_S|} \tag{3.7}$$

where t is a term in R_S, $\beta(t, R_S)$ is 1 if t is a misspelling and 0 otherwise, and $|R_S|$ is a normalization factor representing the length of R_S. We use the Enchant[9] library to identify misspelled terms as it wraps many existing spellchecking libraries, such as Ispell, Aspell, and MySpell.

Lastly, we look at the coverage of misspellings using Equation 3.8.

$$MC(R_s) = \frac{\sum_{t \in R_{Su}} \gamma\left(t, TL_{all}\right)}{\sum_{t \in R_{Su}} \beta\left(t, R_s\right)} \tag{3.8}$$

where $\beta(.)$ is defined as in Equation 3.7, t is a term in R_{Su}, which is the set of unique terms in R_S, TL_{all} is the set of terms resulting from merging the term list for each category in ObjCat, and $\gamma(.)$ evaluates to 1 if t is identified as a misspelling and it occurs in TL_{all}, and 0 otherwise.

Objectionability detection. Based on its effectiveness in similar classification tasks [87], we use the Random Forest model to identify objectionable resources. Using the feature representation of R as input, a trained Random Forest model[10] produces as output a binary probability distribution $\hat{\mathbf{y}}$ over each class–objectionable and not–such that $\hat{\mathrm{y}} \in [0, 1]$ for R. To serve as the sensitivity score exploited by the risk module, we define S_{bad} as the probability value of R being associated with the objectionable class (Equation 3.9).

[9] `https://abiword.github.io/enchant/`

[10] Max leaf node, min leaf samples, and min sample split are set to 32. Max depth is set to 8.

$$S_{bad}(R) = Judge_{bad}(R_S) \tag{3.9}$$

3.2 `REdORank`: From Theory to Practice

`REdORank` is powered by the LTR algorithm, **AdaRank** [127], as it is one of the more prevalent algorithms in LTR research [48, 69, 78, 84]. AdaRank uses a listwise approach (defined in Section 2.1), which is the most effective in terms of ranking accuracy when used for Web search [20, 118].

3.2.1 AdaRank

AdaRank is a boosting algorithm wherein a collection of weakly-defined rankers are linearly combined to create an overall ranker that is more accurate than any of the individual weak rankers. A weak ranker is defined as as $h_t = P_t(i)E(\pi(q_i, \mathbf{d}_i, x_k)$, where $q_i \in Q$ is a set of queries, $d_i \in D_q$ is a ranked list of documents per query, y_i is the ground truth for document d_i, $x_1, ..., x_k$ are the feature representations for each document, E is an evaluation measure, and $P_1(i) = \frac{1}{|Q|}$ is an initial weight.

Given a set of training data $\{q_i, d_i, y_i\}$, AdaRank takes an iterative approach such that at each iteration $t \in T$, a set of weak rankers are initialized with the current weights, a ranking permutation ($\pi(\cdot)$) is predicted and evaluated, and the weights are updated. The pseudocode for this process is outlined in Algorithm 1.

Like all LTR algorithms, AdaRank learns a ranking function through the optimization of an evaluation measure. The metric most commonly-used for optimization is Normalized Discounted Cumulative Gain (**NDCG**) [64, 79]. The goal of NDCG is to measure the agreement between a predicted ranked list and the ground truth

Algorithm 1 AdaRank Algorithm, reproduced from [127].

```
Input: $S = \{(q_i, \mathbf{d}_i, y_i)\}_{i=1}^{m}$, and parameters $E$ and $T$.
Initialize $P_1(i) = \frac{1}{m}$.
**For** $t = 1, \cdots, T$
    • Create weak ranker $h_t$ with weighted distribution $\mathbf{P}_t$ on training data $S$.
    • Choose $\alpha_t$
        $a_t = \frac{1}{2} \cdot ln \frac{\sum_{i=1}^{m} P_t(i)\{1 + E(\pi(q_i, \mathbf{d}_i, h_t), \mathbf{y}_i)\}}{\sum_{i=1}^{m} P_t(i)\{1 - E(\pi(q_i, \mathbf{d}_i, h_t), \mathbf{y}_i)\}}$
    • Ceate $f_t$
        $f_t(\overrightarrow{x}) = \sum_{k=1}^{t} \alpha_k h_k(\overrightarrow{x})$
    • Update $\mathbf{P}_{t+1}$
        $P_{t+1} = \frac{exp\{-E(\pi(q_i, \mathbf{d}_i, f_t), \mathbf{y}_i)\}}{\sum_{j=1}^{m} exp\{-E(\pi(q_j, \mathbf{d}_j, f_t), \mathbf{y}_j)\}}$
**End For**
Output ranking model $f(\overrightarrow{x}) = f_T(\overrightarrow{x}))$.
```

for a query q. The "Gain" in NDCG is the relevance gain, or the benefit of showing relevant resources higher in the ranking. The relevance gain of each resource retrieved in response to a query is determined based on its position in a ranked list, as in Equation 3.10.

$$g_i = 2^{rel_i} - 1 \tag{3.10}$$

where g_i is the relevance gain of the i^{th} resource in a ranked list, and rel_i is the corresponding relevance ground truth.

The "Discount" portion of NDCG is a penalization applied to resources that are relevant, yet they appear lower in a ranked list. It is necessary to ensure that this discount is not too steep, to account for persistent users that are more likely to explore deeper into ranked lists [64]. As such, NDCG employs a logarithmic discount based on a resource's position in a ranked list, as seen in Equation 3.11.

$$d_i = log(rank_i + 1) \tag{3.11}$$

where d_i is the discount for the i^{th} resource in a ranked list, and $rank_i$ is the position of the i^{th} resource in said ranking.

When considering a ranked list in response to a query q, the "Cumulative" aspect comes into play as the accumulation of discounted gains, beginning at the top of the ranked list until a particular position. Formally, this is known as Discounted Cumulative Gain (**DCG**) and is defined as in Equation 3.12.

$$DCG@k(q) = \sum_{i=1}^{k} \frac{g_i}{d_i} \tag{3.12}$$

where k is a cutoff value, i.e., the number of resources examined in a list, i is a position in the ranking, and g_i and d_i are as defined in Equations 3.10 and 3.11, respectively.

Accounting for the need to measure agreement across an entire list where the number of relevant documents may vary, DCG must be "Normalized", resulting in NDCG, calculated as in Equation 3.13.

$$NDCG@k(q) = \sum_{i=1}^{k} \frac{DCG@k(q)}{IDCG@k(q)} \tag{3.13}$$

where k, i, and $DCG@k(q)$ are as defined in Equation 3.12, and $IDCG@k(q)$ is the DCG as calculated for a perfect, i.e., "ideal", ranked list, up to position k.

The benefit of NDCG is its ability to account for various degrees of relevance, due to the manner in which relevance gain is determined. Unlike more traditional counterparts for ranking that examine a single relevance value, `REdORank` considers multiple signals for relevance of a resource, namely the educational alignment and readability. Additionally, NDCG does not account for explicit signals of non-relevance, such as the objectionability of a resource. Therefore, we seek to expand NDCG to account for additional relevance and non-relevance signals.

3.2.2 Multi-Perspective Optimization with Cost Sensitivity

The goal of a search system is to retrieve resources from a collection that have the highest relevance with regards to a user's query. In some cases, these collections contain resources that are not meant to be seen by all users, such as private medical documents or, in the case of a government system, top secret missives. These types of resources are known as sensitive resources. As a way to avoid presenting sensitive materials in response to online inquiries, Sayed and Oard [110] introduced an extended version of the DCG metric, called Cost Sensitive Discounted Cumulative Gain (**CS-DCG**). This new metric (Equation 3.14), introduces a cost penalty, or a risk factor, for displaying a sensitive document within a ranking of retrieved resources.

$$CS - DCG_k = \sum_{i=1}^{k} \frac{g_i}{d_i} - c_i \tag{3.14}$$

where k, g_i, and d_i are defined as in Equation 3.12, and c_i is the sensitivity cost of showing a sensitive document at rank position i.

Incorporating CS-DCG into an LTR model such as AdaRank empowers the model to learn to rank sensitive documents lower than those that are not sensitive. This aligns with what we seek to do with the objectionability perspective of `REdORank`: eradicate from top ranking positions those resources that can be perceived as sensitive for the user group and environment that are the focus of our work. Thus, instead of depending upon the traditional NDCG when training its LTR re-ranker, `REdORank` uses CS-DCG for optimization purposes. In this case, we use as the sensitivity cost c_i S_{bad} (Equation 3.9).

CS-DCG accounts for objectionable resources, but still only considers a single signal for relevance gain. In the context of our work, however, it is imperative to leverage the influence that both educational alignment and readability have into determining the relevance of a given resource. It is not sufficient to simply linearly combine the respective grade level and educational alignment scores, S_{edu} and S_{read}, computed in Sections 3.1.2 and 3.1.1, respectively. Instead, it is important to understand the interdependence between these two scores in terms of dictating relevance gain.

To model the connection between educational alignment and readability we take inspiration from a weighting scheme core to Information Retrieval: TF-IDF. TF (or term frequency) captures the prominence of a term within a resource, whereas IDF (or inverse document frequency) characterizes the "amount of information carried by a term, as defined in information theory" [27] and is computed as a proportion of the size of a collection over the number of resources in the collection in which the term appears. In our case, this weighting scheme acts as a sort of "mixer" for the traits that inform relevance. Intuitively, we treat S_{edu} as representative of the content of R (in terms of matching the classroom setting) and readability as the discriminant factor with respect to resources considered for ranking purposes. Given the often high readability levels of online resources [10, 12], we use 13 as the readability level representative of the collection, and therefore use it as the max readability in the numerator for IDF. With this in mind, the mixer score for R informed by the two aforementioned signals of relevance is computed as in Equation 3.15.

$$mixer(R) = S_{read}(R) \times log_2(\frac{13}{S_{edu}(R)}) \tag{3.15}$$

By incorporating multiple signals of relevance into the determination of relevance gain, and the expansion of DCG with a cost-sensitivity factor, we have defined an updated metric that serves to ensure `REdORank` explicitly learns to respond to the user group, task and environment requirements, by prioritizing resources that align with our user group and environment, while preventing the presentation high in the ranking of retrieved resources that are objectionable for our environment.

CHAPTER 4

EXPERIMENTAL RESULTS

In this chapter, we describe the experiments we conducted in order to answer our research questions. We begin by assessing the the correctness of the readability formula that is part of the design of `REdORank`, as well as the performance of the proposed strategies for detection of educational and objectionable resources. This enable us to show that the methodologies considered to account for each of our perspectives are sound. We then assess the overall design of `REdORank`, via both an ablation study and comparison with baseline counterparts. Along the way, we provide in-depth analysis of the results for each experiment.

4.1 Finding a Readability Formula Fitting Web Resources

There is no readability formula that is the default when estimating the complexity of texts. Thus, it is essential that we empirically examine formulas in an effort to identify the one best suited for determining the level of complexity of Web resources. In our examination, we look at the efficacy of readability formulas for their originally intended purpose: the estimation of reading level of published texts, i.e., books and news articles. We then investigate how the formulas perform when applied to the text snippets of Web resources. Through comparison of the results in each medium, books and Web, we select the formula best suited for our audience and context.

We begin our exploration with traditional readability formulas, as they are simple to compute and are broadly adopted [51, 82]. Traditional formulas also require less data than the machine and deep learning solutions and are freely accessible, supporting the open availability intended for `REdORank`. Initially, we look at (i) *Coleman-Liau Index* (Equation 4.1), as it was designed for digital texts, and to be easily calculated automatically [25], (ii) *Flesch-Kincaid* (Equation 4.2) [66], as it is a well-known formula that has been employed to estimate complexity of Web resources focusing primarily on upper-elementary to secondary grade levels [15], and (iii) *Spache Readability Formula* (Spache for short, Equation 4.3) [113], intended for texts targeted to readers in grades 1–3. The latter relies on a static vocabulary of 1,064 words that are considered easy for children to comprehend. Each of these three formulas were designed to estimate the reading level of published materials, e.g., books or news and magazine articles.

$$\textit{Coleman-Liau}(R) = (0.058 \times |l_R|) - (0.296 \times |s_R|) - 15.8 \tag{4.1}$$

where R is a given resource, $|l_R|$ is the number of letters in R and $|s_R|$ represents the number of sentences in R.

$$\textit{Flesch-Kincaid}(R) = (0.39 \times sl_R) + (11.8 \times spw_R) - 15.59 \tag{4.2}$$

where sl_R represents the average sentence length of R and spw_R represents the number of syllables per word in R.

$$\textit{Spache}(R) = (0.141 \times w_R/s_R) + (0.086 * dif(R)) + 0.839 \tag{4.3}$$

where w_R and s_R are the number of words and sentences in R, respectively. The function $dif(R)$ determines the percentage of difficult words in R, where a word is deemed difficult if it does not appear in the "easy words" vocabulary[1].

Although there are many datasets that can be used to assess the performance of readability formulas, to the best of our knowledge, none was designed for Web resources, our target audience, or labelled specifically with grade levels. With that in mind, we created our own, denoted TextComp that is comprised of 4,860 instances of the form <text sample, grade_label, source>. We explicitly included in TextComp samples of resources from printed and digital mediums allowing us to probe the applicability of different formulas for our target audience and context. Samples in TextComp are distributed as follows:

- 235 book excerpts extracted from the appendices of the CCSS [62], each associated with a range of grade levels. We opt to use the minimum grade level from these ranges as the label, as children reading below their reading level experience less difficulty with comprehension versus when reading above their reading level [7].

- 2,084 books from Reading A-Z (**RAZ**) labeled with their corresponding reading level[2].

- 2,541 Web resources from the Idaho Digital Learning Alliance (**IDLA**), a collection of online course materials serving K-12 students [5], each associated with a grade pre-determined by expert educators.

[1] `https://github.com/cdimascio/py-readability-metrics/blob/master/readability/data/spache_easy.txt`

[2] RAZ uses a 26-letter scale assigned by experts for readability [72]. To enable fair comparison across formulas, these letter labels are mapped grade labels ranging from Kindergarten to 6^{th} grade, using the conversion table provided by RAZ [71].

To quantify the performance of each formula F considered in our exploration, we rely on Error Rate (**ER**), computed as in Equation 4.4. We determine significance of our results using the Kruskal-Wallis H-test [67] with a p<0.05. Unless otherwise stated, all results reported in the rest of this section are significant.

$$ER(F) = \frac{1}{|\textsc{TextComp}|} \sum_{d \in \textsc{TextComp}} |T\hat{C}_{d,F} - TC_d| \tag{4.4}$$

where $|\textsc{TextComp}|$ is the size of $\textsc{TextComp}$, d is an instance in $\textsc{TextComp}$, TC_d is the known grade for d, and $T\hat{C}_{d,F}$ is the grade level of d estimated using F.

To attain a base understanding of how each formula performs when applied to their original target resource, we compute the ER using the 2,319 books in $\textsc{TextComp}$. As shown in Figure 4.1, the Coleman-Liau Index exhibits a lower error rate than Flesch-Kincaid and Spache at the 9^{th} grade level and above. Interestingly, even though it is a commonly-used formula [15], Flesch-Kincaid produces the largest ER across grade level when estimating the complexity of books. Spache is the least error prone for K–6^{th} grade, thus best aligning with our audience (K-4).

To validate if this performance translates to Web resources, we repeat the same experiment using the Web resources in $\textsc{TextComp}$. Much like for books, Spache is the least error prone formula for grades 1–4 & 6–8 (see Figure 4.1). In contrast to books, Flesch-Kincaid fares better for 8^{th} and 10^{th} grade Web resources, with Coleman-Liau performing best for the remaining grades. Outcomes from the presented analysis serve as indication of Spache being the formula best suited for the task at hand: estimating grade levels of Web resources targeting young searchers.

Regardless of its effectiveness for our audience and context, we note that Spache's vocabulary is limited and was last updated in the 1970s. As language changes

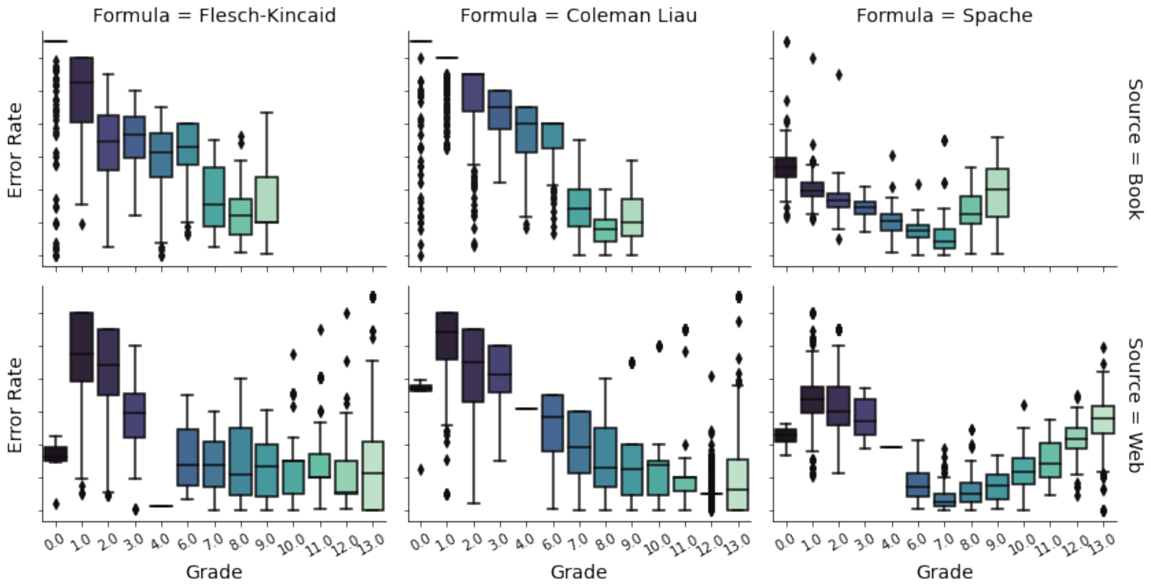
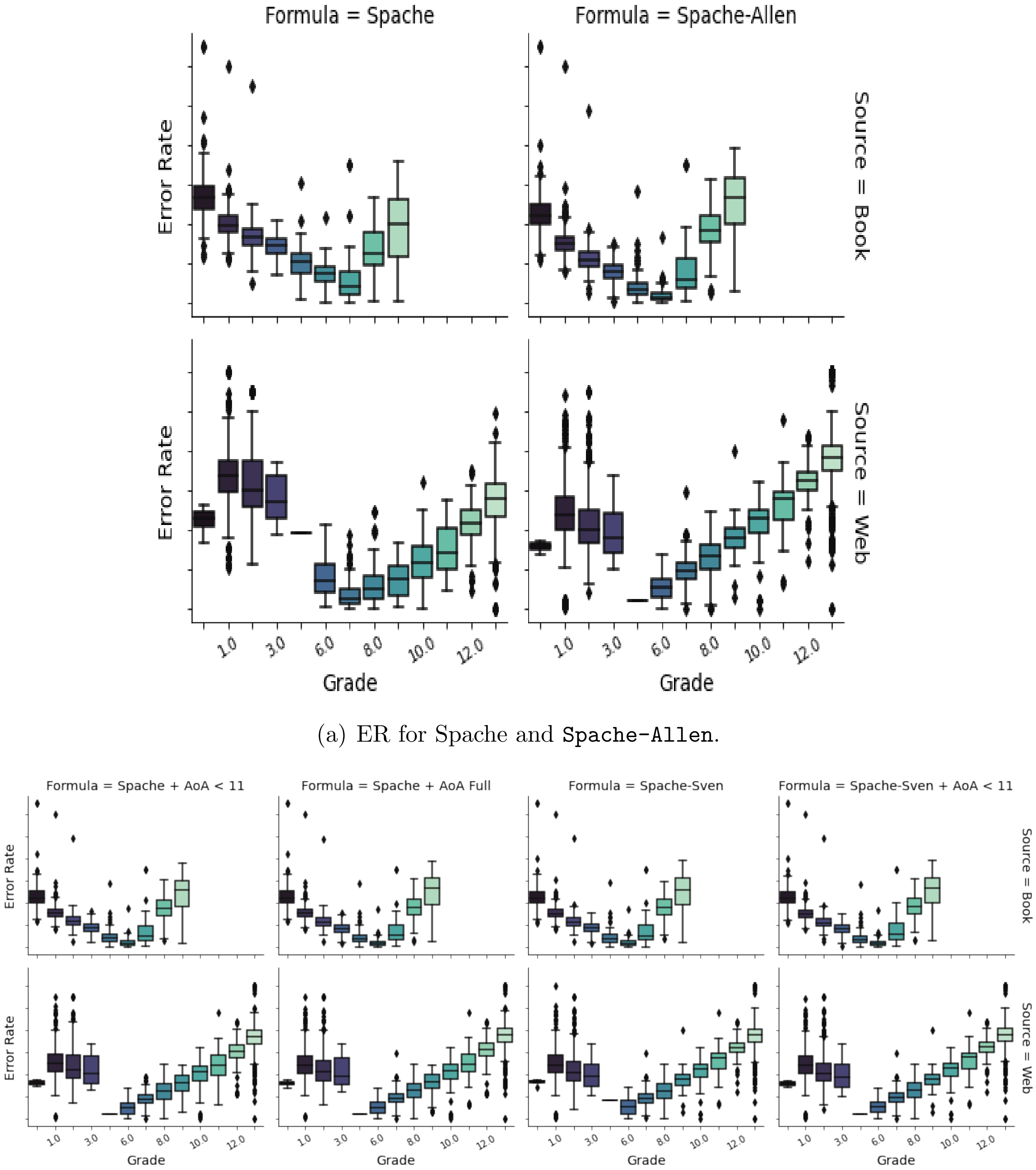


Figure 4.1: ER distribution across different grades for traditional readability formulas.

over time [106], an outdated vocabulary may not capture easy terms for children in today's world, potentially leading the formula to misleadingly estimate the complexity of a text. The New Dale-Chall formula [22] increased the vocabulary considered by the original Dale-Chall formula [31] from 763 to 3,000 in the 1990s seeking to update the formula in response to a new set of passages with assigned grade levels for comparison to determine difficulty of texts, known as "criterion passages", for the development of readability formulas [36]. These insights inspired us to pursue an extended version of Spache's vocabulary. Madrazo Azpiazu et al. [82] already considered enhancing Spache's original vocabulary list, by including a dictionary of 48,000 non-stop lemmatized terms the authors extracted from children-related websites (Sven) as part of the vocabulary considered by the formula. The enhanced formula was successfully used to determine if a query was child-like. However, this enhancement relied on word frequency analysis of child-related websites and assumed that terms added to the vocabulary would be understood by children, which may not always be the case. With the intent of including vocabulary that children learn through instruction, we take advantage of the Age of Acquisition (**AoA**) dataset. This dataset contains acquisition ratings in the form of ages, ranging from 1–17 years, for ∼30,000 English words [68], which we use to augment the original Spache vocabulary, specifically including terms from AoA with an average age of acquisition equal to or below 11 years as this aligns with our target audience. We further examine an augmentation of Spache that expands the original vocabulary with all of the words in AoA, as well as having the AoA words serve as the sole supporting vocabulary. We posit that there is a benefit to simultaneously considering terminology that children have been exposed to through websites as well as terminology that has been taught, therefore we explore the combining of the original Spache vocabulary with the combined terms from AoA and the dictionary from [82].

Following the same experimental procedure used for the comparison of the Flesch-Kincaid, Coleman-Liau, and Spache formulas, we compare the ER of the modified Spache formulas. First, we examine how each performs using book resources in TextComp, the results of which can be seen in Figure 4.2. The Spache formula using the original easy words vocabulary is more error prone than when using the extended vocabularies. While the differences in ER differences are less pronounced for the formulas using the extended vocabularies, when the vocabulary incorporates Sven, the error rate is the lowest for grades K–5^{th}. We attribute this benefit to Sven being specifically tailored to words used by children, whereas Spache's original vocabulary is a list of general terms deemed easy to understand. There is no significant difference in ER when applied to Grade 6 resources. Looking again at Figure 4.2, we see a similar trend for Web resources, with the formula yielding the lowest error rates for resources below the 6^{th} grade being the version of Spache using Sven + AoA

Full(i.e., `Spache-Allen`). From this experiment, we see that expanding the original Spache vocabulary has a positive effect on the ER.



(a) ER for Spache and `Spache-Allen`.

(b) ER for variations of Spache using AoA and Sven.

Figure 4.2: Distribution of ER for Spache-inspired formulas across grades.

Which readability formula simultaneously suits resource type, context needs, and user group outlined for our task? (RQ1) Given our focus is on children in grades K–4th and the significantly reduced ER of the Spache formula with the SVEN + AoA FULL vocabulary, we deem this extended version, defined as Spache-Allen in Equation 3.1, as the most suitable to be included in the design of REdORank.

4.2 Detecting Educational Web Resources

The identification of online resources that align with common learning outcomes for K–4 students among those considered for re-ranking is a key part of REdORank. To do so, we introduced BiGBERT (Section 3.1.2), a deep learning classification model. To ensure that the outcomes of BiGBERT, that directly impact the performance of REdORank, can serve as an effective signal of relevance, we undertake a robust evaluation of BiGBERT, which we discuss below.

4.2.1 Experimental Set-up

To evaluate the performance of BiGBERT we conduct an ablation study to demonstrate that each component–URL vector, snippet vector, and domain knowledge infusement–is necessary for BiGBERT to identify educationally aligned resources. Additionally, we conduct a comparison with similar classification models to contextualize the performance of BiGBERT.

An in-depth exploration of the literature in this area reveals that there is no **dataset** we can use to assess the performance of models that determine the educational alignment of Web resources. Thus, we build one, which we call EDUSITES, using URLs (with text in English) from Alexa Top Sites [6]–based on the well-known Open Directory Project (**ODP**) [23, 92]. We treat as educational the 1,273 URLs in subcategories *Pre-School* and *School Time* from *Kids & Teens*. We also randomly select 3,998 non-educational URLs uniformly distributed among *Adult*, *Business*, *Recreation*, and *Games*. To validate that labels in EduSites align (or not) with our definition of educational, an education expert annotated a representative sample (n = 527). As in [94], we calculate the accuracy between the two annotations (Alexa vs. expert) per sample, obtaining an inter-annotator agreement of 94.7%.

For performance assessment, we use **Accuracy**, a common classification metric, along with False Positive (**FPR**) and False Negative (**FNR**) ratios, to offer insights on the type of misclassified resources. A false-positive is a resource marked as educational, that is not. A false-negative is the opposite, an educational resource marked as non-educational. Significance of results is determined with McNemar's test, $p<0.05$.

To the best of our knowledge, there are no domain-specific classifiers that we can use to contextualize `BiGBERT`'s performance. Thus, we optimize and adapt several classifiers to detect K–4 Web resources:

1. **BoW** [49], a bag-of-words model that computes cosine similarity between a vectorized resource URL and ODP category descriptions to determine the resource's respective category (note that we use the text of learning outcomes from educational standards in lieu of category descriptions).

2. **BGCNN** [105], a model based on a BiGRU with a CNN which identifies child-friendly URLs.

3. **BERT4TC** [130], a text classifier that uses a BERT encoder to perform topic and sentiment classification,.

4. **Hybrid-NB** [1], a hybrid model which examines both URL and content of websites to determine their target audience (i.e., Algerian users). Reported results for BGCNN and BERT4TC are the average of 5-fold cross validation.

Besides the aforementioned classifiers, we explore **variations** of BiGBERT, where **U**, **S**, and **E** indicate when BiGBERT examines only URLs, snippets, and infuses educational information, respectively. Via an ablation study, we showcase the contributions of the URL and snippet vectorizers towards BiGBERT's overall architecture.

4.2.2 Results and Discussion

We offer below an analysis of the results of the experiments conducted to assess the design of BiGBERT. We summarize these results in Table 4.1.

Table 4.1: Performance analysis of BiGBERT (ablation study along with experiments related to comparisons with counterparts). The suffixes **-U** and **-S** indicate model applied to URL and snippet only, resp.; **-E** indicates model augmented with educational data. * and † significant w.r.t. BiGBERT and non-educational counterpart, resp. Significance determined with McNemar's test, p<0.05.

Row	**Type**	**Models**	**Accuracy**	**FPR**	**FNR**
1	Baseline	BoW	.7205 *	.115	.796
2	State-of the-art	BGCNN	.8399 *	.073	.432
3		BERT4TC	.9353 *	.041	.140
4		Hybrid-NB	.8600 *	.145	.123
5	Ablation Study	BiGBERT-U	.8276 *	.073	.484
6		BiGBERT-U-E	.8287 * †	.072	.483
7		BiGBERT-S	.9374 *	.027	.175
8		BiGBERT-S-E	.9334 *	.038	.155
9		BiGBERT-U-S	.9381 *	.035	.146
10		BiGBERT	**.9533 †**	**.027**	**.106**

Reports in [105] showcase the effectiveness of only examining URLs to identify sites as child-friendly. This motivates us to study the applicability of the approach for detecting educational Web resources targeting K–4 populations. The accuracy of BoW does not surpass the 75% mark attained via a naive baseline (one always predicting non-educational due to the unbalanced nature of our dataset). BGCNN, BiGBERT-U, and BiGBERT-U-E outperform more traditional models with accuracy rates in the low 80 percentile. We attribute the increase in performance to the fact that state-of-the-art models do not assume URL token independence, unlike BoW. Results from our analysis indicate that when semantic and context-rich information is available, URLs are a valuable source to inform classification. The number of misclassified educational resources in this case, however, is high. In fact, nearly half of educational samples, which comprise 25% of our data, are labelled non-educational (see respective FNR). This leads us to investigate additional information sources that can contribute to the classification process.

As content analysis is a staple of classification, it is logical to consider knowledge inferred from snippets to better support the classification of K–4 educational Web resources. This is demonstrated by significant performance improvements of Hybrid-NB, BiGBERT-U-S, and BiGBERT over counterparts solely looking at URLs (BoW and BGCNN). BiGBERT significantly outperforms hybrid models in accuracy and FPR. Fewer false positives means lower likelihood for potentially inappropriate sites being labelled educational, which is of special importance given the domain and audience of our work. The results suggest that snippets, combined with URLs, do help identify educational resources. However, the higher FNR of BiGBERT-U-S compared to Hybrid-NB, again points to the misclassification of educational resources. This can be seen on samples like *www. sesamestreet. org*, recognized as educational by

Hybrid-NB but overlooked by BiGBERT-U-S. This would suggest that the lack of explicit domain knowledge is a detriment to BiGBERT-U-S.

The accuracy of BiGBERT increases when using Edu2Vec and fine-tuned BERT embeddings (rows 9 vs 10 in Table 4.1). To determine whether the improvement is the result of explicitly infusing educational knowledge into the classification process, we compare BiGBERT-U and BiGBERT-S with educationally-augmented counterparts. Our experiments reveal a significant decrease in FPR and FNR between BiGBERT-U and BiGBERT-U-E; non significant between BiGBERT-S and BiGBERT-S-E. Unlike for URL variations, BiGBERT-S-E's performance improved only in FNR after augmentation. We attribute this to the relatively small training set used for fine-tuning in comparison to the initial pre-training set for BERT, leading to less new contextual information learned by the standard transformer model. Nonetheless, the significant increases in accuracy and decreases in FPR and FNR for BiGBERT when compared to BiGBERT-U-S suggest that domain-specific knowledge can have a positive effect on the classification of educational resources. This is illustrated by the URL *www. xpmath. com*, a site to support math education in grades 2^{nd}–9^{th}, that is labelled non-educational by BiGBERT-U-S, yet it is correctly recognized as educational by BiGBERT.

Do snippets along with URLs help identify educational resources? Does domain-specific knowledge affect identification of educational resources (RQ2) From the results presented thus far, it emerges that indeed both URLs and snippets are required to adequately portray the educational alignment of resources. Moreover, by explicitly infusing domain-knowledge into the design of BiGBERT, the range of educational resources identified expands as compared to simply using traditional BERT. Overall, in light of the success BiGBERT has at identifying K–4 educational resources while minimizing false negatives and false positives, we deem BiGBERT as suitable to be included in the design of REdORank. By optimizing on the output of BiGBERT as a relevance signal, REdORank is able to address our environment and audience.

4.3 Identifying Objectionable Resources

REdORank is designed to demote Web resources that are objectionable for children in the classroom context. Given that REdORank relies on $Judge_{bad}$ (introduced in Section 3.1.3) to identify these types of resources, it is imperative to verify its reliability to avoid error propagation. Thus, we undertake an in-depth analysis of performance, which we discuss below.

4.3.1 Experimental Set-up

To the best of our knowledge, there does not exist a labelled dataset with coverage for all categories within ObjCat, therefore we construct one: ObjSet. This dataset, extracted from the Alexa Top Sites directory, is comprised of 10,006 samples of the form <snippet, URL, label>, where label is 1 for objectionable samples, and 0 otherwise. We treat as objectionable 2,096 resources for which their corresponding Alexa category name contains as a substring one of the ObjCat category and sub-category names. The remaining 7,910 additional resources from Alexa serve as non-objectionable counterparts. By selecting non-objectionable resources in a roughly 4:1 ratio to objectionable, we simulate a real world setting where objectionable resources will make up a smaller portion of SERP.

To measure performance, we use **Accuracy**, **FPR**, and **FNR**. In this case, a false-positive is a resource marked as objectionable but is not. A false-negative is the opposite, an objectionable resource marked as non-objectionable. Further, we compare and contrast $Judge_{bad}$ with that of a number of counterpart models, each adopting a different strategy for identifying objectionable resources. Through this comparison we gain insights and contextualize how $Judge_{bad}$ performs with respect to existing solutions.

- **MNB**. A bag-of-words Multinomial Naive-Bayes model that computes the TF-IDF for the resource descriptions provided by ODP to determine the resource's respective class.

- **BERT4TC** [130]. A text classifier that uses the state-of-the-art BERT encoder coupled with a multi-layer perceptron to perform topic and sentiment classification.

- **AWESSOME** [1]. A framework that combines the VADER sentiment lexicon [61] with BERT to predict the sentiment intensity of sentences.

- **KSAppropriateness** [87]. Focused on the same user group and environment, this model leverages a curated lexicon to analyze term frequency and term proportion within the content of a Web resource determine appropriateness.

To ensure a fair comparison among models, ObjSet is divided into a training and test set using an 80/20 split and all snippets are pre-processed in the same manner: tokenized, punctuation removed, and lemmatized. Significance of results is determined with McNemar's test, $p<0.05$.

4.3.2 Results and Discussion

As shown in Table 4.2, $Judge_{bad}$ achieves an accuracy score of 84.9%–a rate that exceeds the expected performance of a majority classifier (79%) given the data distribution in ObjSet. Probing deeper on $Judge_{bad}$'s performance, we observe that, interestingly, $Judge_{bad}$ has a higher FPR than FNR. We attribute the false negatives, i.e., those that are marked as non-objectionable when they are objectionable, to the shorter length of some ObjCat lexicons, as such a lexicon provides less possibility of clear representation for a given category. This is visible upon manual inspection by resources related to alcohol and tobacco, a sub-category of Drugs, being mislabelled as non-objectionable. The Drugs lexicon has a total of 100 terms, of which only 21 relate to alcohol and tobacco. A manual inspection of missed objectionable resources reveals that in some situations $Judge_{bad}$ correctly identifies resources as objectionable that have a label of non-objectionable, but are not appropriate for children in a classroom. For example, consider `www.kids-in-mind.com`, an online platform providing reviews of the content of films (gore, adult language, nudity, etc.) so parents can make decisions about what films to show their children, or `www.casinocity.com`, a directory site of casino and gambling related reviews and games. Both samples are labeled as non-objectionable as per Alexa's descriptions, yet, neither is a site suitable for children in a classroom setting. From cases like this, we argue that $Judge_{bad}$ is accomplishing its intended goal.

To contextualize $Judge_{bad}$'s performance, we compare it to that of several related models. MNB exhibited very similar performance to $Judge_{bad}$, in terms of FPR and FNR. Interestingly, there a significant difference in the accuracy of the two models. Delving into which resources were misclassified, there are some differences

Table 4.2: Evaluation of objectionable classification models using ObjSet. * indicates significance w.r.t. $Judge_{bad}$ determined by McNemar's test with Bonferroni Correction, p<0.05.

Model	Accuracy	FPR	FNR
Judge_{bad}	0.849	0.038	0.574
MNB	0.856*	0.002	0.679
BERT4TC	0.209*	1.0	0.0
AWESSOME	0.209	1.0	0.0
KSAppropriateness	0.209	1.0	0.0

to be found. Manually inspecting the misclassifications, we found that MNB tends to misidentify websites known to contain sexual content. We connect these misses to the lack of vocabulary depth present in a bag-of-words model. Surprisingly, BERT4TC, AWESSOME, and KSAppropriateness act as a minority classifier, always assigning objectionable labels to resources. With the 4:1 distribution of ObjSet, the expectation was for BERT4TC to learn to classify non-objectionable materials. We ascribe the unexpected result to overfitting, in the sense that the model learned representations of the terms in objectionable resources as clear identifiers regardless of their presence or context in non-objectionable counterparts. With BERT serving as an internal component for AWESSOME, we similarly attribute this overfitting behavior to AWESSOME. We ascribe the performance of KSAppropriateness to the combination of reduced features considered (only looking at sexually explicit and hate speech terms) and the uneven distribution of the data.

Can topic-specific lexicons empower the identification of objectionable Web resources? (RQ3) The use of extended lexicons on topics inappropriate for a classroom, beyond pornography and hate speech, provide additional lenses for $Judge_{bad}$ to discern what is and is not objectionable. Overall, given the demonstrated performance of $Judge_{bad}$, we deem it suitable to include in the design of REdORank.

4.4 Re-ranking Web Resources with REdORank

Thus far, we have shown the applicability of solutions for identifying the educational alignment, readability, and objectionability of resources. Using these perspectives together, REdORank seeks to re-rank resources to support children searching in a classroom environment by balancing the evidence of resources being inappropriate, and therefore a risk, with the evidence that resources are educational and readable, and therefore rewarding or beneficial. To validate that the design of REdORank answers our research question and thus meets this goal, we undertake a comprehensive evaluation which we discuss in the remainder of this section.

4.4.1 Experimental Set-up

There exist datasets for the evaluation of ranking models based on LTR, such as the MQ2007 and MQ2008 sets [102] or the OHSUMED set [58]. Unfortunately, there is no LTR dataset comprised of queries, resources, and "ideal" labels pertaining to our user group and environment. In addition, none of the existing datasets include known objectionable resources, which are a must in order to explicitly assess the validity of REdORank's design. In light of these two facts, we construct our own dataset RankSet.

The construction of datasets for information retrieval tasks often follows the Cranfield paradigm [124]. For ranking tasks, this process involves beginning with known "ideal" resources. The title of each resource is used as a query to trigger the retrieval of other resources in order to produce a ranked list. The ideal resource is always positioned at the top of the ranking, as it is treated as the ground truth. The remaining top-N ranked resources (excluding the one originating the search, if available) are used to complete the ranked list. The Cranfield paradigm enables the construction of RankSet to ensure an ideal resource is in the top position for every query. However, REdORank also aims to push objectionable resources lower in the rankings. To enable evaluation of this aspect of REdORank, we append at the bottom of the list a known "bad" resource.

To act as the ideal resources for RankSet, we use a collection of 9,540 articles with known reading levels and educational value targeted for children on a variety of topics from NewsELA [91]. For bad resources, we turn to ObjSet (Section 4.3.1). Following the Cranfield paradigm, we use the ideal article titles as queries and using Google's API we retrieve up to 20 resources, their titles, search snippets, and rank positions (we drop queries that lead to no resources or resources with missing content). We assign relevance labels of 2 to the ideal resources, 0 to the known "bad" resources, and 1 to all other resources retrieved from Google. This results in RankSet containing a total of 2,617 queries and 46,881 resources.

To demonstrate the correctness of REdORank's design and its applicability, we undertake an ablation study. REdORank utilizes AdaRank as the underlying LTR algorithm with the expanded CS-DCG metric for optimization. To validate and examine how (i) the expansion of the optimization metric from the more traditional NDCG, and (ii) the incorporation of objectionability as a sensitivity cost, affect its overall performance, we compare REdORank to AdaRank optimized with the standard NDCG metric. Each model is configured with variations that utilize each perspective as standalone features. To further contextualize the performance of REdORank, we perform a comparison with a two other models: (i) LambdaMART, a popular listwise

LTR model that utilizes Multiple Additive Regression Trees [46], with the overall ranking function being the linear combination of regression trees, and *(ii)* Korsce [87], a model designed to rank resources that align with 3^{rd} to 5^{th} grade educational curriculum, are comprehensible for children in that same grade range, are objective in content (i.e., not based in opinion), and are appropriate for the classroom, described in detail in Section 2.2. We treat LambdaMART as a baseline, whereas Korsce (matching our user group, environment, and context) is a state-of-the-art counterpart.

To measure performance, we use NDCG@10 and Mean Reciprocal Rank (**MRR**). MRR seeks to spotlight the average ranking position of the first relevant item. In our case, we find it particularly important to position objectionable resources very low among retrieved results. Therefore, we also compute an alternative version of MRR, in which rather than accounting for the first relevant (ideal) item, we account for the position of the first objectionable item. We call this MRR_{Bad}, where a lower value indicates better performance. Significance of results is verified using a two-tailed student t-test with $p<0.05$; all results reported and discussed in the rest of this section are significant unless stated otherwise.

4.4.2 Results and Discussion

We begin our evaluation of adapting LTR to children searching in the classroom by looking at how a known listwise LTR algorithm, AdaRank, optimized for a standard ranking metric (NDCG), performs when trained to rank according to our chosen perspectives, educational alignment, readability, and objectionability. We train variations of AdaRank with each perspective, educational alignment, readability, and objectionability each acting as a single feature. We refer to these variations with the suffixes -E, -R, and -O, respectively. We train the same set of variations for `REdORank` with the addition of ones that use the mixer (described in Section 3.2.2) to combine the educational alignment and readability perspectives into a single feature. We refer to these with the suffixes -M, where the mixed values are the only feature, and -MER where the mixed values are used alongside the individual perspectives. Results of the experiments are presented in Tables 4.3 and 4.4.

We first look at each individual perspective as a feature for AdaRank. As anticipated, AdaRank-O performed the worst, as seen by the lower NDCG and MRR scores as well as the higher MRR$_{Bad}$. We attribute this to the fact that AdaRank-O is optimizing for the "risk" perspective, and thus learning to potentially prioritize the known bad resource above the known ideal. When optimizing on the "reward" perspectives, AdaRank-E and AdaRank-R perform better than AdaRank-O. These models place objectionable resources around the 10^{th} position according to MRR$_{Bad}$, while ranking the ideal ones around the 5^{th} position, according to MRR (Rows 1–3 in Table 4.3). This is indicative of these models learning to focus on the types of resources well-suited for our user group and environment. When putting all of the features together, AdaRank outperforms each of the individual variations, indicating the value of each perspective in determining relevance.

So far, we have showcased that the design choices for considering risk and reward perspectives in a re-ranking task are well-founded. However, we surmise that the AdaRank models are learning to rank objectionable resources lower as a beneficial side-effect of optimizing on the educational alignment and readability. To account for objectionable as an explicit signal of cost, and to balance that risk with the reward of the other perspectives, we turn to REdORank, optimized for nCS-DCG.

For REdORank-E, REdORank-R, and REdORank-O, we see similar performances to those of their AdaRank counterparts (Rows 5–7 and 2–4 in Table 4.3, respectively).

Table 4.3: Performance of REdORank and ablation variations using RankSet. The suffixes -R, -E, -O indicate Readability only, Educational only, and Objectionable only, respectively. -M indicates the use of the mixer for educational alignment and readability, and -MER indicates the use of the mixer *with* -E and -R. * indicates significance w.r.t. REdORank and bold indicates best performing for each metric.

Row	Algorithm	Optimization Metric	NDCG	MRR	$\textbf{MRR}_{Bad}$
1	AdaRank	NDCG	0.778*	0.226*	0.097*
2	AdaRank-E	NDCG	0.765*	0.209	0.110*
3	AdaRank-R	NDCG	0.774*	0.222	0.101*
4	AdaRank-O	NDCG	0.675*	0.148*	0.537*
5	REdORank-E	nCS-DCG	0.765*	0.209	0.110*
6	REdORank-R	nCS-DCG	0.774*	0.222	0.101*
7	REdORank-O	nCS-DCG	0.675*	0.148*	0.537*
8	REdORank-M	nCS-DCG	0.765*	0.209	0.110*
9	REdORank-MER	nCS-DCG	0.777	0.218	**0.089***
10	REdORank	nCS-DCG	**0.779**	**0.228**	0.097

This further highlights that the perspectives matter. We posit that the interconnection of educational alignment and readability will serve as a beneficial composite signal for the relevance of a resources. For this reason, we utilize the mixer described in Section 3.2.2 to combine the two perspectives. Surprisingly, REdORank-M performs worse in all metrics when compared to REdORank-R, and performs the same as REdORank-E. To fully investigate whether this combined perspective could provide value to the re-ranking, we created REdORank-MER. Lending credence to the idea of incorporating a combined perspective, REdORank-MER outperformed each of the individual perspective variations. While this variation performed significantly better than REdORank in terms of MRR_{Bad}, it performed worse for the other two metrics. This highlights that the explicit consideration of a sensitivity cost factor, alongside multiple perspectives of relevance, has beneficial affects on re-ranking resources for children searching in the classroom.

Table 4.4: Performance of REdORank and baselines using RankSet. * indicates significance w.r.t.REdORank and bold indicates best performing for each metric.

Algorithm	**Optimization Metric**	**NDCG**	**MRR**	**MRR$_{Bad}$**
LambdaMART	NDCG	**0.784**	**0.228**	**0.081***
Korsce	N/A	0.753*	0.209	0.163*
REdORank	nCS-DCG	0.779	**0.228**	0.097

The results so far have shown that the design for REdORank is well-founded. To attain a better understanding of how REdORank performs, we also compare it to both a state-of-the-art counterpart, Korsce, and a baseline LTR algorithm, in LambdaMART. The results of these two models ranking the resources in RankSet can be seen in Table 4.4. We see that REdORank performs significantly better than Korsce for all metrics. This is visually represented in Figure 4.3. We attribute the difference in performance to the fact that Korsce ranks in a pointwise, weighted objective manner. That is, for each resource, each perspective score is multiplied by an empirically determined weight, and then added together to create the ranking score. In contrast, REdORank learns a single dynamic weight that accounts for each perspective simultaneously as opposed to individually. LambdaMART learns to rank by optimizing on pairwise comparisons of documents. Surprisingly, LambdaMART performs significantly better than REdORank for the RankSet. While this was unexpected, as listwise LTR algorithms have been shown to be more effective when applied to Web search [20, 118], we attribute the discrepancy in performance to the structure of the dataset. RankSet only contains a single ideal resource, which a pairwise algorithm is more likely to "locate" by nature of directly comparing documents. On the other hand, REdORank is more likely to miss the ideal resource as it does not explicitly compare each resource to every other one, but rather considers their relevance in a relative manner within the list. In real-world scenarios, where more than one ideal resource is likely to be in a single list, a listwise approach is better suited to the re-ranking task.

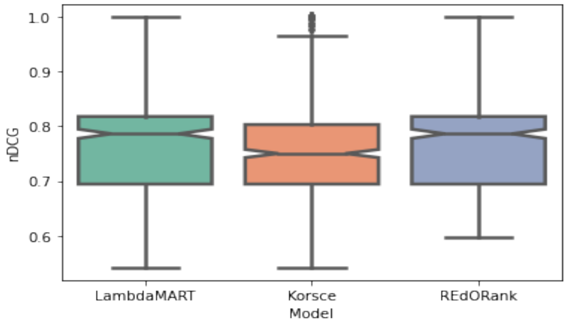


Figure 4.3: NDCG@10 for different re-ranking models using RankSet.

Does the adaptation of an LTR model to account for multiple perspectives lead to the prioritization of resources that are relevant to both children and the classroom setting? (RQ4) Given its visibly higher lower bound on NDCG@10 over its counterparts (as seen in Figure 4.3), its successful performance regarding ranking known educational and readable resources high in the rankings, and its expected generalizability to real-world re-ranking scenarios, we consider the design of `REdORank` to be an appropriate model for providing re-ranking to search systems supporting children's search activity in the classroom.

CHAPTER 5

CONCLUSION

The work presented in this thesis advances Information Retrieval for Children–centered on the design, development, and assessment of strategies that enable children's online information discovery. This research area is of particular importance due to the ubiquitousness of search engines (SE), and the fact that children are considered non-traditional users, as the vast majority of the academic and commercial efforts in this area target adults. Given the broad range of children's search skills when it comes to searching [50], and the wide range of inquiries children turn to SE for, particularly with the ongoing COVID-19 pandemic, we explicitly scoped our work to focus on children ages 6–11 using Web search tools in the classroom context. To better support this user group, we presented a novel re-ranking strategy, `REdORank`, that serves as training wheels for facilitating children's identification of resources that are relevant to their information needs.

Responding to findings reported in the literature regarding the manner in which children search, e.g., their propensity to fixate on top-ranked results [50, 53], as well as the manner in which SE respond to children's queries [13, 12], e.g., offering resources children cannot comprehend, `REdORank` examines resources retrieved by commercial SE–preferred by children–and prioritizes them in a manner that those best suited for the context and user group at hand are ranked higher. To do so, `REdORank` appraises resources based on three perspectives: educational alignment, readability, and objectionability. By combining of all of these perspectives, REdORank ensures that SE can better respond to children's search behavior by balancing the risk and reward value of resource content.

With educational alignment, the goal is to prioritize resources that support learning. Even if resources are educational, for their content to be of value, it must be readable and therefore comprehensible. To that end, REdORank looks at the readability of resources. Unfortunately, not all resources online are meant for, or even appropriate for, children to engage with in a classroom context. As such, REdORank incorporates the objectionability perspective so it can position such resources lower in the ranking where they are less likely to be seen and engaged with by children.

An in-depth analysis of REdORank revealed that a multi-perspective LTR model is an effective solution to re-rank resources for children in the classroom. Through the experiments conducted, we can conclude that the deliberate inclusion of perspectives connected to a particular user group and environment can benefit the performance of a model in the re-ranking of resources retrieved from a mainstream SE. In fact, explicitly considering signals of "negative" perspectives, and balancing them with "positive" signals of relevance provides significant improvements in performance. Integral to REdORank's ability to re-rank resources, we discovered that URLs and snippets provided an effective proxy for the classification of an online resource as educational, or not. Based on the ablation study conducted with REdORank, we also found that snippets are useful in the identification of resources as objectionable. Finally, from the readability standpoint, we noticed that considering a vocabulary more carefully tailored to contemporary language and adjusted to children's age of acquisition of terms can improve estimation of readability for online resources targeted towards children.

As a result of an extensive empirical exploration to determine the validity of each of the perspectives informing REdORank's design, the following additional contributions emerged: BiGBERT, a classification model to identify educationally aligned Web resources through an ensemble deep learning approach; and $Judge_{bad}$, a lexicon-based classification model for identifying online resources objectionable for children in a classroom context. In addition to these classification models, we introduce a new readability formula that is effective at estimating the readability of online resources.

Through the course of our work, a number of limitations and pathways for further research came to light. Spache-Allen is designed and evaluated only on its applicability to estimate the readability of English language resources. However, the Web is world-wide and covers many different domains of information. Therefore, performing a similar empirical exploration involving different domains, e.g., legal or medical, as well as multilingual readability formulas may provide further valuable insights for the many areas of research interested in text understanding, particularly Natural Language Processing (NLP). REdORank leverages readability as an internal feature, based on Spache-Allen, which only looks at text resources to estimate their readability. In the future, we plan to expand this perspective to consider other methods of readability estimation that account for the presence of additional media elements, e.g., images and charts on web pages. Another limitation is the lack of consideration of a users' prior knowledge on a subject. Future work investigating the connection between pre-existing topical knowledge and readability estimation can bridge this gap and further align supporting tools such as REdORank with their target user groups.

When exploring objectionable resources, we followed existing state-of-the-art approaches and treated all categories in ObjCat as unquestionably objectionable. However, children do not necessarily require a one-size-fits all solution. For instance, content that is objectionable for a 4^{th} grader may not be so for a 12^{th} grader. We recognize this as a limitation of $Judge_{bad}$, and suggest increasing granularity of identifying what is and is not objectionable for children of various ages.

Additional limitations involve RankSet, the dataset used for assessing the proposed re-ranking model, which is constructed based on the Cranfield paradigm. As such, the dataset is limited to only a single ideal resource in response to each query. Unfortunately, having only a single ideal resource is not indicative of real world SERP, thus leading us to pursue further studies on the performance of `REdORank` in a realistic environment. With that, immediate next steps include a user study involving the examination of children's search behavior when using a search system with and without `REdORank`.

Outcomes from this thesis have implications for researchers investigating children's Web search. `REdORank` is a step towards adapting mainstream SE to classroom use, yet still it focuses only on specific perspectives to inform relevance gain. It is worth researching the benefits of combining additional relevance signals, from sources beyond the text of a resource, such as where the resource comes from, or who wrote it. Such factors contribute to the credibility of a resource. Unfortunately, children are known to not judge the credibility of online resources [55], making credibility a valuable extra perspective to bring into the fold for re-rankers. This can be achieved quickly and effectively using the mixer strategy employed by `REdORank` (Section 3.2.2) which enables the simultaneous aggregation of multiple scores into a single one. While this mixer is currently used for "reward" perspectives, it can be replicated for "risk" perspectives as well. For example, children struggle to identify if something is misinformation or not [99]. As a result, they may consider a misinformation resource as credible. By extending the cost to include perspectives beyond objectionability, such as misinformation, REdORank can prioritize resources that are grounded in fact.

Ongoing research in Human-Computer Interaction has explored how visual elements of a SERP affect children's interactions with results [3, 4]. REdORank can provide further avenues of exploration regarding how to identify the type of resources and visual elements that can serve as visual clues. For example, a small book icon with a number inside can be added alongside the display of a result on a traditional SERP to indicate the reading level of a resource. Similarly, an icon representing a small schoolhouse could be added to indicate that a resource is of educational value.

Through the addition of visual elements rooted in the traits considered during the ranking process, transparency of search systems can increase. Users can be offered a glimpse behind the curtain into how a particular system works. This can impact the ease of use and understandability of a system. Additionally, such visual elements can benefit users *learning to search* by providing, over time, a visible connection between ranked resources and the query that searchers can use to improve query formulation.

REdORank is a tangible step towards designing an adaptive search tool for children. Perhaps most impactful, is the possibility of using REdORank to support the *searching to learn* portion of the search as learning paradigm. Searching to learn is the act of seeking information to gain new knowledge within an educational setting [13, 108], which aligns very well with the purpose of REdORank given the effectiveness for identifying, and propensity to rank higher, Web resources of educational value.

REFERENCES

[1] Ouessai Abdessamed and Elberrichi Zakaria. Web site classification based on url and content: Algerian vs. non-algerian case. In *2015 12th International Symposium on Programming and Systems (ISPS)*, pages 1–8. IEEE, 2015.

[2] Claudia Margarita Acuña-Soto, Vicente Liern, and Blanca Pérez-Gladish. A vikor-based approach for the ranking of mathematical instructional videos. *Management Decision*, 2019.

[3] Mohammad Aliannejadi, Monica Landoni, Theo Huibers, Emiliana Murgia, and Maria Soledad Pera. Children's perspective on how emojis help them to recognise relevant results: Do actions speak louder than words? In *Proceedings of the 2021 Conference on Human Information Interaction and Retrieval*, pages 301–305, 2021.

[4] Garrett Allen, Benjamin L Peterson, Dhanush Kumar Ratakonda, Mostofa Najmus Sakib, Jerry Alan Fails, Casey Kennington, Katherine Landau Wright, and Maria Soledad Pera. Engage!: Co-designing search engine result pages to foster interactions. In *Interaction Design and Children*, pages 583–587, 2021.

[5] Idaho Digital Learning Alliance. Idaho digital learning portal. `https://portal.idiglearning.net/K12/CourseCatalog`, 2021. (accessed January 19, 2021).

[6] Inc Amazon. Alexa top sites. `https://www.alexa.com/topsites/category`, 2020. (accessed September 17, 2020).

[7] Steven J Amendum, Kristin Conradi, and Meghan D Liebfreund. The push for more challenging texts: An analysis of early readers' rate, accuracy, and comprehension. *Reading Psychology*, 37(4):570–600, 2016.

[8] Steven J Amendum, Kristin Conradi, and Elfrieda Hiebert. Does text complexity matter in the elementary grades? a research synthesis of text difficulty and elementary students' reading fluency and comprehension. *Educational Psychology Review*, 30(1):121–151, 2018.

[9] Jonathan Anderson. Lix and rix: Variations on a little-known readability index. *Journal of Reading*, 26(6):490–496, 1983.

[10] Hélder Antunes and Carla Teixeira Lopes. Readability of web content. In *2019 14th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–4. IEEE, 2019.

[11] Oghenemaro Anuyah, Ion Madrazo Azpiazu, and Maria Soledad Pera. Using structured knowledge and traditional word embeddings to generate concept representations in the educational domain. In *Companion Proceedings of The 2019 World Wide Web Conference*, pages 274–282, 2019.

[12] Oghenemaro Anuyah, Ashlee Milton, Michael Green, and Maria Soledad Pera. An empirical analysis of search engines' response to web search queries associated with the classroom setting. *Aslib Journal of Information Management*, 2019.

[13] Ion Madrazo Azpiazu, Nevena Dragovic, Maria Soledad Pera, and Jerry Alan Fails. Online searching and learning: Yum and other search tools for children and teachers. *Information Retrieval Journal*, 20(5):524–545, 2017.

[14] Rebekah George Benjamin. Reconstructing readability: Recent developments and recommendations in the analysis of text difficulty. *Educational Psychology Review*, 24(1):63–88, 2012.

[15] Dania Bilal. Comparing google's readability of search results to the flesch readability formulae: a preliminary analysis on children's search queries. *Proceedings of the American Society for Information Science and Technology*, 50(1):1–9, 2013.

[16] Dania Bilal and Meredith Boehm. Towards new methodologies for assessing relevance of information retrieval from web search engines on children's queries. *Qualitative and Quantitative Methods in Libraries*, 2(1):93–100, 2017.

[17] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit.* O'Reilly Media, Inc., 2009.

[18] Sebastian Bruch, Shuguang Han, Michael Bendersky, and Marc Najork. A stochastic treatment of learning to rank scoring functions. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pages 61–69, 2020.

[19] Christopher JC Burges. From ranknet to lambdarank to lambdamart: An overview. *Learning*, 11(23-581):81, 2010.

[20] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136, 2007.

[21] David Carmel, Elad Haramaty, Arnon Lazerson, and Liane Lewin-Eytan. Multi-objective ranking optimization for product search using stochastic label aggregation. In *Proceedings of The Web Conference 2020*, pages 373–383, 2020.

[22] Jeanne Sternlicht Chall and Edgar Dale. *Readability revisited: The new Dale-Chall readability formula*. Brookline Books, 1995.

[23] Wanyu Chen, Fei Cai, Honghui Chen, and Maarten De Rijke. Personalized query suggestion diversification in information retrieval. *Frontiers of Computer Science*, 14(3):143602, 2020.

[24] Benjamin Clavié and Kobi Gal. Edubert: Pretrained deep language models for learning analytics. *arXiv preprint arXiv:1912.00690*, 2019.

[25] Meri Coleman and Ta Lin Liau. A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, 60(2):283, 1975.

[26] Kevyn Collins-Thompson, Paul N Bennett, Ryen W White, Sebastian De La Chica, and David Sontag. Personalizing web search results by reading level. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 403–412, 2011.

[27] W Bruce Croft, Donald Metzler, and Trevor Strohman. *Search engines: Information retrieval in practice*, volume 520. Addison-Wesley Reading, 2010.

[28] Washington Cunha, Sérgio Canuto, Felipe Viegas, Thiago Salles, Christian Gomes, Vitor Mangaravite, Elaine Resende, Thierson Rosa, Marcos André Gonçalves, and Leonardo Rocha. Extended pre-processing pipeline for text classification: On the role of meta-feature representations, sparsification and selective sampling. *Information Processing & Management*, 57(4):102263, 2020.

[29] Washington Cunha, Vítor Mangaravite, Christian Gomes, Sérgio Canuto, Elaine Resende, Cecilia Nascimento, Felipe Viegas, Celso França, Wellington Santos Martins, Jussara M Almeida, et al. On the cost-effectiveness of neural and non-neural approaches and representations for text classification: A comprehensive comparative study. *Information Processing & Management*, 58 (3):102481, 2021.

[30] Xinyi Dai, Jiawei Hou, Qing Liu, Yunjia Xi, Ruiming Tang, Weinan Zhang, Xiuqiang He, Jun Wang, and Yong Yu. U-rank: Utility-oriented learning to rank with implicit feedback. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 2373–2380, 2020.

[31] Edgar Dale and Jeanne S Chall. A formula for predicting readability: Instructions. *Educational research bulletin*, pages 37–54, 1948.

[32] Judith H Danovitch. Growing up with google: How children's understanding and use of internet-based devices relates to cognitive development. *Human Behavior and Emerging Technologies*, 1(2):81–90, 2019.

[33] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[34] Yajuan Duan, Long Jiang, Tao Qin, Ming Zhou, and Heung Yeung Shum. An empirical study on learning to rank of tweets. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 295–303, 2010.

[35] Sergio Duarte Torres and Ingmar Weber. What and how children search on the web. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 393–402, 2011.

[36] William H DuBay. *Smart Language: Readers, Readability, and the Grading of Text.* ERIC, 2007.

[37] Benjamin Edelman. Empirical analysis of google safesearch. *Berkman Center for Internet & Society, Harvard Law School*, 14, 2003.

[38] Carsten Eickhoff, Pavel Serdyukov, and Arjen P de Vries. Web page classification on child suitability. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, pages 1425–1428, 2010.

[39] Michael D Ekstrand, Katherine Landau Wright, and Maria Soledad Pera. Enhancing classroom instruction with online news. *Aslib Journal of Information Management*, 2020.

[40] Ahmed Elnaggar, Christoph Gebendorfer, Ingo Glaser, and Florian Matthes. Multi-task deep learning for legal document translation, summarization and multi-label classification. In *Proceedings of the 2018 Artificial Intelligence and Cloud Computing Conference*, pages 9–15, 2018.

[41] Vladimir Estivill-Castro and Alessandro Marani. Towards the ranking of webpages for educational purposes. In *CSEDU (1)*, pages 47–54, 2019.

[42] Vanessa Figueiredo and Eric M Meyers. The false trade-off of relevance for safety in children's search systems. *Proceedings of the Association for Information Science and Technology*, 56(1):651–653, 2019.

[43] Elizabeth Foss, Allison Druin, Robin Brewer, Phillip Lo, Luis Sanchez, Evan Golub, and Hilary Hutchinson. Children's search roles at home: Implications for designers, researchers, educators, and parents. *Journal of the American Society for Information Science and Technology*, 63(3):558–573, 2012.

[44] W Nelson Francis and Henry Kucera. Brown corpus manual. *Letters to the Editor*, 5(2):7, 1979.

[45] Thomas François and Eleni Miltsakaki. Do nlp and machine learning improve traditional readability formulas? In *Proceedings of the First Workshop on Predicting and Improving Text Readability for target reader populations*, pages 49–57, 2012.

[46] Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[47] Wolfe Garbe. Symspell. `https://github.com/wolfgarbe/SymSpell`, 2020.

[48] N Geetha and PT Vanathi. Knowledge transfer for efficient cross domain ranking using adarank algorithm. *International Journal of Business Intelligence and Data Mining*, 14(1-2):89–105, 2019.

[49] Filippo Geraci and Tiziano Papini. Approximating multi-class text classification via automatic generation of training examples. In *International Conference on Computational Linguistics and Intelligent Text Processing*, pages 585–601. Springer, 2017.

[50] Tatiana Gossen and Andreas NüRnberger. Specifics of information retrieval for young users: A survey. *Information Processing & Management*, 49(4):739–756, 2013.

[51] Kelsey Leonard Grabeel, Jennifer Russomanno, Sandy Oelschlegel, Emily Tester, and Robert Eric Heidel. Computerized versus hand-scored health literacy tools: a comparison of simple measure of gobbledygook (smog) and flesch-kincaid in printed patient education materials. *Journal of the Medical Library Association: JMLA*, 106(1):38, 2018.

[52] Jiafeng Guo, Yixing Fan, Liang Pang, Liu Yang, Qingyao Ai, Hamed Zamani, Chen Wu, W Bruce Croft, and Xueqi Cheng. A deep look into neural ranking models for information retrieval. *Information Processing & Management*, 57 (6):102067, 2020.

[53] Jacek Gwizdka and Dania Bilal. Analysis of children's queries and click behavior on ranked results and their thought processes in google search. In *Proceedings of the 2017 conference on conference human information interaction and retrieval*, pages 377–380, 2017.

[54] Karl Gyllstrom and Marie-Francine Moens. Wisdom of the ages: toward delivering the children's web with the link-based agerank algorithm. In *Proceed-*

ings of the 19th ACM international conference on Information and knowledge management, pages 159–168, 2010.

[55] Elina K Hämäläinen, Carita Kiili, Miika Marttunen, Eija Räikkönen, Roberto González-Ibáñez, and Paavo HT Leppänen. Promoting sixth graders' credibility evaluation of web pages: an intervention study. *Computers in Human Behavior*, 110:106372, 2020.

[56] Mahdi Hashemi. Web page classification: a survey of perspectives, gaps, and future directions. *Multimedia Tools and Applications*, pages 1–25, 2020.

[57] Samer Hassan and Rada Mihalcea. Learning to identify educational materials. *ACM Transactions on Speech and Language Processing (TSLP)*, 8(2):1–18, 2008.

[58] William Hersh, Chris Buckley, TJ Leone, and David Hickam. Ohsumed: An interactive retrieval evaluation and new large test collection for research. In *SIGIR'94*, pages 192–201. Springer, 1994.

[59] Anett Hoppe, Peter Holtz, Yvonne Kammerer, Ran Yu, Stefan Dietze, and Ralph Ewerth. Current challenges for studying search as learning processes. *Proceedings of Learning and Education with Web Data, Amsterdam, Netherlands*, 2018.

[60] Mark Hughes, Irene Li, Spyros Kotoulas, and Toyotaro Suzumura. Medical text classification using convolutional neural networks. *Studies in Health Technology and Informatics*, 235:246–50, 2017.

[61] Clayton Hutto and Eric Gilbert. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 8, 2014.

[62] Common Core State Standards Initiative. Appendix b: Text exemplars and sample performance tasks, 2020. URL `http://www.corestandards.org/assets/Appendix_B.pdf`.

[63] Common Core State Standards Initiative. Common core state standards for english language arts & literacy in history/social studies, science, and technical subjects, 2020. URL `http://www.corestandards.org/wp-content/uploads/ELA\_Standards1.pdf`.

[64] Kalervo Järvelin and Jaana Kekäläinen. Ir evaluation methods for retrieving highly relevant documents. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.*, pages 41–48. ACM, 2000.

[65] Zenun Kastrati, Ali Shariq Imran, and Sule Yildirim Yayilgan. The impact of deep learning on document classification using semantically rich representations. *Information Processing & Management*, 56(5):1618–1632, 2019.

[66] J Peter Kincaid, Robert P Fishburne Jr, Richard L Rogers, and Brad S Chissom. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. Technical report, Naval Technical Training Command Millington TN Research Branch, 1975.

[67] William H Kruskal and W Allen Wallis. Use of ranks in one-criterion variance analysis. *Journal of the American statistical Association*, 47(260):583–621, 1952.

[68] Victor Kuperman, Hans Stadthagen-Gonzalez, and Marc Brysbaert. Age-of-acquisition ratings for 30,000 english words. *Behavior research methods*, 44(4): 978–990, 2012.

[69] Saar Kuzi, Sahiti Labhishetty, Shubhra Kanti Karmaker Santu, Prasad Pradip Joshi, and ChengXiang Zhai. Analysis of adaptive training for learning to rank in information retrieval. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 2325–2328, 2019.

[70] Monica Landoni, Davide Matteri, Emiliana Murgia, Theo Huibers, and Maria Soledad Pera. Sonny, cerca! evaluating the impact of using a vocal assistant to search at school. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 101–113. Springer, 2019.

[71] Inc. LAZEL. Level correlation chart. `https://www.readinga-z.com/learninga-z-levels/level-correlation-chart/`, 2021. (accessed January 18, 2021).

[72] Inc. LAZEL. Reading a-z: The online reading program with downloadable books to print and assemble. `https://www.readinga-z.com/`, 2021. (accessed January 18, 2021).

[73] Lung-Hao Lee, Yen-Cheng Juan, Hsin-Hsi Chen, and Yuen-Hsien Tseng. Objectionable content filtering by click-through data. In *Proceedings of the 22nd ACM*

international conference on Information & Knowledge Management, pages 1581–1584, 2013.

[74] Lukas Lerche. *Using implicit feedback for recommender systems: characteristics, applications, and challenges.* PhD thesis, Technischen Universität Dortmund, 2016.

[75] Dirk Lewandowski. Evaluating the retrieval effectiveness of web search engines using a representative query sample. *Journal of the Association for Information Science and Technology*, 66(9):1763–1775, 2015.

[76] Hang Li. A short introduction to learning to rank. *IEICE TRANSACTIONS on Information and Systems*, 94(10):1854–1862, 2011.

[77] Junjie Liang, Jinlong Hu, Shoubin Dong, and Vasant Honavar. Top-n-rank: A scalable list-wise ranking method for recommender systems. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 1052–1058. IEEE, 2018.

[78] Hendi Lie, Darren Lukas, Jonathan Liebig, and Richi Nayak. A novel learning-to-rank method for automated camera movement control in e-sports spectating. In *Australasian Conference on Data Mining*, pages 149–160. Springer, 2018.

[79] Tie-Yan Liu. *Learning to rank for information retrieval.* Springer Science & Business Media, 2011.

[80] Fan Ma, Haoyun Yang, Haibing Yin, Xiaofeng Huang, Chenggang Yan, and Xiang Meng. Online learning to rank in a listwise approach for information

retrieval. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1030–1035. IEEE, 2019.

[81] Craig Macdonald, Rodrygo LT Santos, and Iadh Ounis. The whens and hows of learning to rank for web search. *Information Retrieval*, 16(5):584–628, 2013.

[82] Ion Madrazo Azpiazu, Nevena Dragovic, Oghenemaro Anuyah, and Maria Soledad Pera. Looking for the movie seven or sven from the movie frozen? a multi-perspective strategy for recommending queries for children. In *Proceedings of the 2018 Conference on Human Information Interaction & Retrieval*, pages 92–101, 2018.

[83] Alessandro Marani. Webedurank: an educational ranking principle of web resources for teaching. In *ICWL Doctoral Consortium*, pages 25–36. Citeseer, 2016.

[84] Ryan McBride, Ke Wang, Zhouyang Ren, and Wenyuan Li. Cost-sensitive learning to rank. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4570–4577, 2019.

[85] Changping Meng, Muhao Chen, Jie Mao, and Jennifer Neville. Readnet: A hierarchical transformer framework for web article readability analysis. *Advances in Information Retrieval*, 12035:33, 2020.

[86] Yu Meng, Yunyi Zhang, Jiaxin Huang, Chenyan Xiong, Heng Ji, Chao Zhang, and Jiawei Han. Text classification using label names only: A language model self-training approach. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 2020.

[87] Ashlee Milton, Oghenemaro Anuyah, Lawrence Spear, Katherine Landau Wright, and Maria Soledad Pera. A ranking strategy to promote resources supporting the classroom environment. In *Proceedings of the 2020 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT'20)*, 2020.

[88] Eleni Miltsakaki. Matching readers' preferences and reading skills with appropriate web texts. In *Proceedings of the Demonstrations Session at EACL 2009*, pages 49–52, 2009.

[89] Hamid Mohammadi and Seyed Hossein Khasteh. Text as environment: A deep reinforcement learning text readability assessment model. *arXiv preprint arXiv:1912.05957*, 2019.

[90] Michinari Momma, Alireza Bagheri Garakani, Nanxun Ma, and Yi Sun. Multi-objective ranking via constrained optimization. In *Companion Proceedings of the Web Conference 2020*, pages 111–112, 2020.

[91] Newsela. Newsela article corpos, 2016. URL `https://newsela.com/data`.

[92] Shastri L Nimmagadda, Dengya Zhu, and Amit Rudra. Knowledge base smarter articulations for the open directory project in a sustainable digital ecosystem. In *Companion Proceedings of the International Conference on World Wide Web*, pages 1537–1545, 2017.

[93] Jorge Nocedal and Stephen Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

[94] Stefanie Nowak and Stefan Rüger. How reliable are annotations via crowdsourcing: a study about inter-annotator agreement for multi-label image annotation. In *Proceedings of the international conference on Multimedia information retrieval*, pages 557–566, 2010.

[95] Harrie Oosterhuis and Maarten de Rijke. Policy-aware unbiased learning to rank for top-k rankings. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 489–498, 2020.

[96] Zohreh Ovaisi, Ragib Ahsan, Yifan Zhang, Kathryn Vasilaky, and Elena Zheleva. Correcting for selection bias in learning-to-rank systems. In *Proceedings of The Web Conference 2020*, pages 1863–1873, 2020.

[97] Liang Pang, Jun Xu, Qingyao Ai, Yanyan Lan, Xueqi Cheng, and Jirong Wen. Setrank: Learning a permutation-invariant ranking model for information retrieval. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 499–508, 2020.

[98] Deepshikha Patel and Prashant Kumar Singh. Kids safe search classification model. In *2016 International Conference on Communication and Electronics Systems (ICCES)*, pages 1–7. IEEE, 2016.

[99] Jodi Pilgrim and Sheri Vasinda. Fake news and the “wild wide web”: A study of elementary students’ reliability reasoning. *Societies*, 11(4):121, 2021.

[100] Marina A Hoshiba Pimentel, Israel Barreto Sant’Anna, and Marcos Didonet Del Fabro. Searching and ranking educational resources based on terms clustering. In *ICEIS (1)*, pages 507–516, 2018.

[101] KR Premlatha and TV Geetha. Re-ranking of educational materials based on topic profile for e-learning. In *2012 International Conference on Recent Trends in Information Technology*, pages 217–221. IEEE, 2012.

[102] Tao Qin and Tie-Yan Liu. Introducing letor 4.0 datasets. *arXiv preprint arXiv:1306.2597*, 2013.

[103] R Rajalakshmi and Chandrabose Aravindan. A naive bayes approach for url classification with supervised feature selection and rejection framework. *Computational Intelligence*, 34(1):363–396, 2018.

[104] R Rajalakshmi, Hans Tiwari, Jay Patel, Ankit Kumar, and R Karthik. Design of kids-specific url classifier using recurrent convolutional neural network. *Procedia Computer Science*, 167:2124–2131, 2020.

[105] R Rajalakshmi, Hans Tiwari, Jay Patel, R Rameshkannan, and R Karthik. Bidirectional gru-based attention model for kid-specific url classification. In *Deep Learning Techniques and Optimization Strategies in Big Data Analytics*, pages 78–90. IGI Global, 2020.

[106] Christian Ramiro, Mahesh Srinivasan, Barbara C Malt, and Yang Xu. Algorithms in the historical emergence of word senses. *Proceedings of the National Academy of Sciences*, 115(10):2323–2328, 2018.

[107] Deborah K Reed and Sarah Kershaw-Herrera. An examination of text complexity as characterized by readability and cohesion. *The Journal of Experimental Education*, 84(1):75–97, 2016.

[108] Soo Young Rieh, Kevyn Collins-Thompson, Preben Hansen, and Hye-Jung Lee. Towards searching as a learning process: A review of current perspectives and future directions. *Journal of Information Science*, 42(1):19–34, 2016.

[109] Javier Sanz-Rodriguez, Juan Manuel Manuel Dodero, and Salvador Sánchez-Alonso. Ranking learning objects through integration of different quality indicators. *IEEE transactions on learning technologies*, 3(4):358–363, 2010.

[110] Mahmoud F Sayed and Douglas W Oard. Jointly modeling relevance and sensitivity for search among sensitive content. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 615–624, 2019.

[111] Avi Segal, Kobi Gal, Guy Shani, and Bracha Shapira. A difficulty ranking approach to personalization in e-learning. *International Journal of Human-Computer Studies*, 130:261–272, 2019.

[112] Dou Shen, Zheng Chen, Qiang Yang, Hua-Jun Zeng, Benyu Zhang, Yuchang Lu, and Wei-Ying Ma. Web-page classification through summarization. In *Proceedings of the 27th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 242–249, 2004.

[113] George D Spache. The spache readability formula. *Good reading for poor readers*, pages 195–207, 1974.

[114] T Sreenivasulu, R Jayakarthik, and R Shobarani. Web content classification techniques based on fuzzy ontology. In *Intelligent Computing and Innovation on Data Science (Proceedings of ICTIDS 2019)*, pages 189–197. Springer, 2020.

[115] Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. How to fine-tune bert for text classification? In *China National Conference on Chinese Computational Linguistics*, pages 194–206. Springer, 2019.

[116] Krysta M Svore, Maksims N Volkovs, and Christopher JC Burges. Learning to rank with multiple objective functions. In *Proceedings of the 20th international conference on World wide web*, pages 367–376, 2011.

[117] Rohail Syed and Kevyn Collins-Thompson. Optimizing search results for human learning goals. *Information Retrieval Journal*, 20(5):506–523, 2017.

[118] Niek Tax, Sander Bockting, and Djoerd Hiemstra. A cross-benchmark comparison of 87 learning to rank methods. *Information processing & management*, 51 (6):757–772, 2015.

[119] T. Tieleman and G. Hinton. Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning, 2012.

[120] Arif Usta, Ismail Sengor Altingovde, Ibrahim Bahattin Vidinli, Rifat Ozcan, and Özgür Ulusoy. How k-12 students search for learning? analysis of an educational search engine log. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pages 1151–1154, 2014.

[121] Arif Usta, Ismail Sengor Altingovde, Rifat Ozcan, and Ozgur Ulusoy. Learning to rank for educational search engines. *IEEE Transactions on Learning Technologies*, 2021.

[122] Joost van Doorn, Daan Odijk, Diederik M Roijers, and Maarten de Rijke. Balancing relevance criteria through multi-objective optimization. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 769–772, 2016.

[123] Nicholas Vanderschantz and Annika Hinze. How kids see search: A visual analysis of internet search engines. In *HCI 2017*. BISL, 2017.

[124] Ellen M Voorhees. The evolution of cranfield. In *Information Retrieval Evaluation in a Changing World*, pages 45–69. Springer, 2019.

[125] Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. Listwise approach to learning to rank: theory and algorithm. In *Proceedings of the 25th international conference on Machine learning*, pages 1192–1199, 2008.

[126] Tian Xia. Support vector machine based educational resources classification. *International Journal of Information and Education Technology*, 6(11):880, 2016.

[127] Jun Xu and Hang Li. Adarank: a boosting algorithm for information retrieval. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 391–398, 2007.

[128] Liu Yang, Qingyao Ai, Damiano Spina, Ruey-Cheng Chen, Liang Pang, W Bruce Croft, Jiafeng Guo, and Falk Scholer. Beyond factoid qa: Effective methods for non-factoid answer sentence retrieval. In *European Conference on Information Retrieval*, pages 115–128. Springer, 2016.

[129] Sevgi Yigit-Sert, Ismail Sengor Altingovde, Craig Macdonald, Iadh Ounis, and Özgür Ulusoy. Explicit diversification of search results across multiple dimensions for educational search. *Journal of the Association for Information Science and Technology*, 2020.

[130] Shanshan Yu, Jindian Su, and Da Luo. Improving bert-based text classification with auxiliary sentence and domain knowledge. *IEEE Access*, 7:176600–176612, 2019.

[131] Junta Zeniarja, Ramadhan Rakhmat Sani, Ardytha Luthfiarta, Heru Agus Susanto, Erwin Yudi Hidayat, Abu Salam, and Leonardus Irfan Bayu Mahendra. Search engine for kids with document filtering and ranking using naive bayes classifier. In *2018 International Seminar on Application for Technology of Information and Communication*, pages 560–564. IEEE, 2018.

[132] Wenjie Zhao, Gaoyu Zhang, George Yuan, Jun Liu, Hongtao Shan, and Shuyi Zhang. The study on the text classification for financial news based on partial information. *IEEE Access*, 8:100426–100437, 2020.

APPENDIX A

DATA FOR EXPERIMENTS

- AoA – Collection of vocabulary terms labelled with the ages at which children acquire understanding of them [68]. Used in design of `Spache-Allen` (Section 3.1.1).

- EduSites – Collection of online resources extracted from Alexa's Top Sites by Category directory that are educational for children (Section 4.2).

- ObjSet – Collection of online resources extracted from Alexa's Top Sites by Category directory that are objectionable for children, i.e., resources that fall within the categories described in Section 4.3.

- RankSet – Collection of ranked search results in response to queries sourced from NewsELA article titles (Section 4.4).

- Sven + AoA Full – Combination of the terms in AoA and Sven. Used in design of `Spache-Allen` (Section 3.1.1).

- Sven – Dictionary of terms extracted from children-related websites for the work done in [82]. Used in design of `Spache-Allen` (Section 3.1.1).

- TextComp – Collection of book and web resources labelled with a reading level in the form of a grade at which a child is able to read the corresponding resource (Section 4.1).