Teach Machine How to Read: Reading Behavior Inspired Relevance Estimation

Xiangsheng Li†, Jiaxin Mao†, Chao Wang‡, Yiqun Liu†*, Min Zhang†, Shaoping Ma†
†Department of Computer Science and Technology, Institute for Artificial Intelligence,
Beijing National Research Center for Information Science and Technology,
Tsinghua University, Beijing 100084, China
‡6ESTATES PTE LTD, Singapore
yiqunliu@tsinghua.edu.cn

ABSTRACT

Retrieval models aim to estimate the relevance of a document to a certain query. Although existing retrieval models have gained much success in both deepening our understanding of information seeking behavior and constructing practical retrieval systems (e.g. Web search engines), we have to admit that the models work in a rather different manner than how humans make relevance judgments. In this paper, we aim to reexamine the existing models as well as to propose new ones based on the findings in how human read documents during relevance judgment. First, we summarize a number of reading heuristics from practical user behavior patterns, which are categorized into implicit and explicit heuristics. By reviewing a variety of existing retrieval models, we find that most of them only satisfy a part of these reading heuristics. To evaluate the effectiveness of each heuristic, we conduct an ablation study and find that most heuristics have positive impacts on retrieval performance. We further integrate all the effective heuristics into a new retrieval model named Reading Inspired Model (RIM). Specifically, implicit reading heuristics are incorporated into the model framework and explicit reading heuristics are modeled as a Markov Decision Process and learned by reinforcement learning. Experimental results on a large-scale public available benchmark dataset and two test sets from NTCIR WWW tasks show that RIM outperforms most existing models, which illustrates the effectiveness of the reading heuristics. We believe that this work contributes to constructing retrieval models with both higher retrieval performance and better explainability.

KEYWORDS

Reading behavior; Retrieval model; Reinforcement learning

ACM Reference Format:

Xiangsheng Li, Jiaxin Mao, Chao Wang, Yiqun Liu, Min Zhang, Shaoping Ma. 2019. Teach Machine How to Read: Reading Behavior Inspired Relevance Estimation. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '19), July 21–25, 2019, Paris, France.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3331184.3331205

*Corresponding author

ACM ISBN 978-1-4503-6172-9/19/07.
https://doi.org/10.1145/3331184.3331205

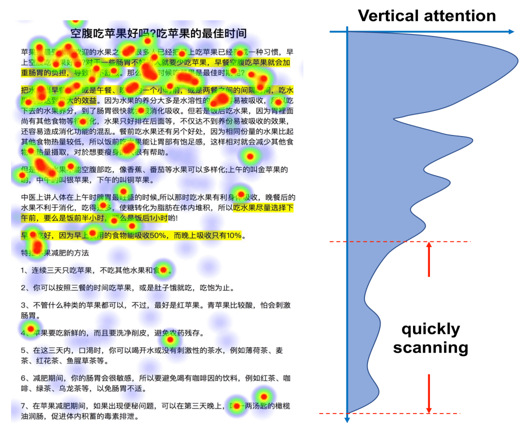


Figure 1: An example of user reading pattern during relevance judgement.

1 INTRODUCTION

Reading behavior during relevance judgement is often inconsistent with general reading behaviors [16]. Users have many particular reading patterns when acquiring knowledge to satisfy their information needs. Based on a public available reading behavior dataset [16], we show an example of user reading pattern during relevance judgement in Figure 1, associated with the distribution of dwell time at each vertical position of a Web page. We can observe three intuitive patterns from this figure: a) More attention is paid to the content at top positions and it decays monotonically towards the bottom of the page (which is also observed in [16]). b) Reading attention is selectively allocated in a document rather than uniformly allocated. Specifically, users selectively read sentences which appears to be important (e.g. potentially meeting user information needs) and skip seemingly irrelevant ones. c) Once users have a confident relevance judgement based on already read content, they tend to speed up the reading process by skimming or even stopping reading before the end part of the document. These patterns provide valuable insights for us to understand the process of actual users' relevance judgement.

Retrieval models are proposed to estimate user's perceived relevance for a certain query-document pair. Therefore, understanding user behavior in relevance judgment can provide valuable implications and heuristics for the designing of retrieval models. Many empirical studies [6, 22] show that good retrieval performance is closely related to the inspiration of actual user behavior, which shows the potential of improving retrieval performance.

However, many existing retrieval models mainly focus on the matching signals between query and document but ignore the heuristics that are inherent in users' reading behaviors. For example, representation-based models [11, 12] integrate query and document information into representation vectors but ignore fine-grained information (e.g., passage or sentence-level relevance). They also assume that users will pay equal attention to different parts in the document, which violates the finding that users' reading attention has a strong position bias [16]. As for interaction models, most of them [11, 32] make a strong assumption that sentences in a document are independent of each other. This is inconsistent with users' sequential reading behavior [16]. In addition, it is demonstrated that users' reading attention decay vertically during the reading process [16], but it is ignored by most existing retrieval models.

While it is important and necessary to consider users' practical reading patterns in designing retrieval models, until now, not many studies have systematically investigated these patterns. The study in [16] is closest to our efforts. It reveals some important findings but does not take a further step to design better retrieval models. In addition, whether the heuristics derived from reading patterns can benefit retrieval performance remains to be investigated. Therefore, in this paper, we aim to integrate the study of users' reading heuristics with retrieval models. We first investigate users' reading patterns during relevance judgment based on a public available reading behavior dataset and propose six heuristics that people usually follow while making relevance judgement. It is found that most existing retrieval models only follow a part of the proposed reading heuristics. We further group the six reading heuristics into implicit and explicit categories, and incorporate them into a novel retrieval model in different ways. Specifically, the explicit heuristics are incorporated into the retrieval model with a reinforcement learning framework while the implicit ones are directly modeled as integral parts of the model. Experimental results on a large-scale public available benchmark dataset (QCL [36]) and two test sets from NTCIR WWW tasks [14, 19] illustrate that most of the proposed reading heuristics have positive impacts on retrieval performance and the newly proposed retrieval model which integrates all of the effective reading heuristics outperforms most existing retrieval models. The main contributions of our work are three folds:

(1) We investigate actual users' reading patterns during relevance judgement and propose six reading heuristics. A number of existing retrieval models are reviewed and compared using these reading heuristics.
(2) We empirically validate these reading heuristics and propose a new Reading Inspired retrieval Model (RIM) according to the effective heuristics with a reinforcement learning framework.
(3) We show that RIM outperforms most existing models in a large-scale benchmark dataset and two NTCIR test sets.

2 RELATED WORK

2.1 User Reading Behaviors

Reading is a complex cognitive process which is originally studied in cognitive psychology by collecting users' eye movement data [24]. Specifically, eye movement is composed of a sequence of fixations (relatively stationary on a point for a period of time) and saccades (rapid scanning between two fixations). In cognitive psychology, there has been a number of reading models elaborating the information acquisition during the reading process. For example, EZ Reader [25] defines different cognitive stages that consider word identification, visual processing, attention, and control of the oculomotor system as joint determinants of eye movement in the reading process. These reading models provide insights into the understanding of the individual's general reading behavior.

However, the reading behavior in information retrieval is often inconsistent with general reading behaviors [16]. Li et al [16] discovered that users' reading process is generally from top to bottom and reading attention is not equally distributed in a document but decays monotonically from top to bottom. When users read text, they often skip seemingly irrelevant information and selectively reading sentences which appear to be important [33]. Michael et al [10] explained it as a tradeoff between the precision of language understanding and attention effort. Similarly, once users have a clear understanding based on already read content, they tend to speed up reading by skimming or even stop reading before the end of the document [27]. Many empirical studies show that good retrieval models [6, 22] are closely related to the inspiration of user behaviors. Thus, we argue that it is important and necessary to consider these heuristics for achieving good retrieval performance.

2.2 Retrieval Models

Existing retrieval models can be categorized into two kinds: statistic probability models and deep models. Statistic probability models such as TF-IDF [3], BM25 [26] and SDM [1], mainly focus on the query frequency in a document. Deep models have gained increasing attention for its ability to automatically learn features from raw text of query and document [17]. Specifically, representation based models [11] aim to build a good representation of query and document and interaction models [6, 9, 11, 22, 32] aim to build local interactions between query and document, and then aggregate each interaction to learn a complex pattern for relevance. ARC-II [11] maps the word embeddings of query and document to an aggregated embedding by CNN. DRMM [9] applies matching histogram mapping to consider query term importance. Kernel pooling in KNRM [32] provides effective multi-level soft matches between query and document. But these models only focus on the matching and ignore users' reading behavior patterns during relevance judgement. DeepRank [22], based on query centric assumption [31], selectively considers the matching occurring at query centric context. HiNT [6] sequentially models passage-level information and accumulates to final relevance, which works like users' sequential reading behavior. However, DeepRank and HiNT only use a few reading heuristics. Different from existing retrieval models, we systematically investigate users' reading patterns and incorporate more reading heuristics into retrieval models.

2.3 Reinforcement Learning in Reading Models

Reinforcement learning is a good approach to model Markov Decision Process [35]. Yu et al. [33] proposed a LSTM-skip model which selectively skips irrelevant information to speed up the computation. Zhang et al. [35] utilized reinforcement learning to select important and task-relevant words in a sentence, which is formulated as a sequential decision problem. These models are based on the selective attention mechanism during the reading process [35]. Furthermore, Liu et al. [18], inspired by the cognitive process of text reading, proposed to early make classification decision and discard the rest text. Yu et al. [34] integrated a model based on users' skip reading behavior, early stop reading behavior and re-reading behavior on text classification task. Fu et al. [8] set a maximal forward step and a maximal backward step, making the model read in two directions like human reading process. In our work, we also apply reinforcement learning to model explicit reading heuristics, where *Selective attention* and *Early stop reading* are considered as action policies in Markov Decision Process.

Table 1: Summary of heuristics from users' reading patterns, where *a*-*d* and *e*-*f* are considered as implicit and explicit reading heuristics, respectively.

#	Heuristic	Description	Implication for retrieval models
a	**Sequential reading**	Reading direction is from top to bottom	The presented order of the content may affect relevance
b	**Vertical decaying attention**	Reading attention is decaying vertically	Retrieval model should assign more weights to the text at the beginning of documents
c	**Query centric guidance**	Reading attention is higher in the contexts around query terms	Retrieval models should follow IR heuristics [7] and capture the interactions between query and document
d	**Context-aware reading**	Reading behavior is influenced by the relevance perception from previously read text	The local relevance of the text should also depend on its surrounding context
e	**Selective attention**	Users will skip some seemingly irrelevant text during relevance judgement	Retrieval models can ignore the text that has no or little influence on relevance
f	**Early stop reading**	Users will stop reading once the read text is enough to make relevance judgement	Retrieval models should be able to estimate the relevance without processing the whole document

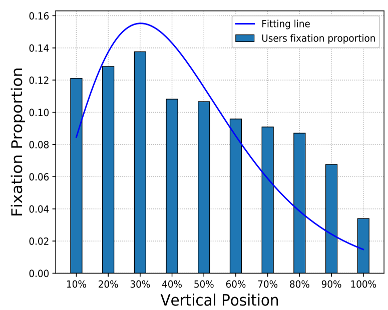


Figure 2: Users' fixation proportion at each vertical position and its fitting line.

3 READING HEURISTICS IN RELEVANCE JUDGEMENT

In this section, we summarize the heuristics derived from users' reading patterns. The analysis is based on a public available reading behavior dataset that was constructed by Li et al. [16] to investigate how users distribute their reading attention during relevance judgement. The dataset contains 29 users' eye-tracking logs when making relevance judgment for 60 documents. By analyzing users' eye movements, Li et al. elaborated the process of users' relevance judgement. Based on this dataset, we further investigate some reading heuristics that are potentially important for the design of retrieval models. We list 6 reading heuristics in Table 1. They are categorized into implicit and explicit types in terms of different implementations in retrieval models, as detailed in Section 4.

3.1 Sequential reading

By calculating the average first arrival time of each vertical position, Li et al. [16] found that users' reading direction is generally from the top position to the bottom of a document. Sequential reading is one of the most obvious patterns in users' reading behavior, which indicates that the content presented *order* may affect users' relevance judgement [5]. We attempt to change the order as inverse order and random order to investigate the necessity of incorporating the sequential reading heuristic in retrieval models.

3.2 Vertical decaying attention

It is found that users' reading attention is not equally distributed in a document [16]. Figure 2 shows users' fixation proportion at each vertical position. We can observe that users' reading attention decreases significantly as the position goes down, which illustrates that the position bias affects users' reading behavior and attention distribution during relevance judgment. The vertical decaying attention heuristic suggests that the retrieval model should assign more weights to the text at the beginning of documents. To incorporate this reading heuristic into the retrieval model, we utilize a Gamma distribution to fit the vertical distribution of eye fixations. The fitted curve is shown as a blue line in Figure 2 and will be used as a decaying coefficient for the RIM model proposed in Section 4.

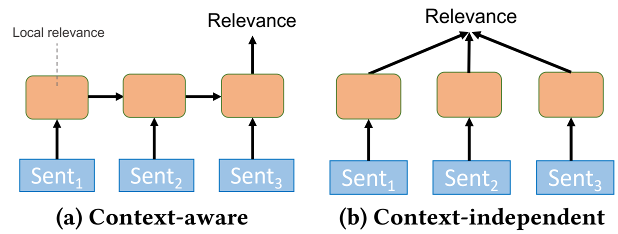


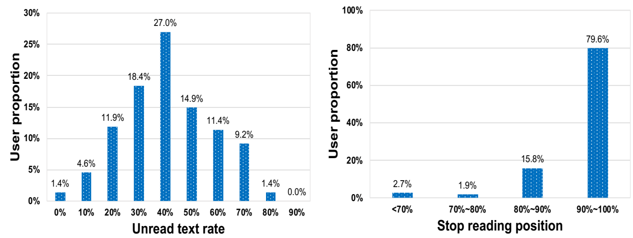
Figure 3: Comparison of context-aware processing and context-independent processing.

3.3 Query centric guidance

Users' reading attention is guided by the search intent, which is reflected by the issued query. Users' reading attention is significantly higher in the context around the query terms [16]. Thus, the text in such query-centric context may play a more important role in determining the relevance of the document. We utilize the similarity between query and target text (e.g., sentence, passage) to model the query centric guidance. The basic idea is to follow IR heuristics and qualify them into retrieval models [7, 17]. Specifically, exact query matching is based on query centric assumption [31] and semantic matching is based on Term Semantic Frequency Constraint [17].

3.4 Context-aware reading

Most retrieval models [9, 11, 32] simply assume that each piece of text is independent of each other, which violates users' reading behavior pattern. Users have different reading behaviors (e.g., reading speed, reading attention) after they perceived different relevance in the context [16]. Thus, the context-aware reading heuristic indicates that it is necessary to consider the contextual influence in the retrieval model. Figure 3 shows a comparison between the context-aware and context-independent relevance model. In context-aware model, the local relevance of each sentence is dependent on its surrounding context. However, context-independent model produces local relevance only relying on a single sentence.



(a) Selective attention **(b) Early stop reading**

Figure 4: User proportion in different situations.

3.5 Selective attention

Due to the tradeoff between the precision of language understanding and attention effort [10], users tend to instinctively select important text to read and skip seemingly irrelevant information. Based on the reading behavior data [16], we calculate the user proportion of different unread text rate in Figure 4 (a). It can be observed that most users do not read full documents while judging relevance, which illustrates that users will selectively read important text and skip seemingly irrelevant ones. It suggests that retrieval models can ignore the text that has no or little influence on relevance.

3.6 Early stop reading

Similarly, once users have a clear understanding based on already read content, they tend to speed up the reading by skimming or even early stopping reading before the end part of the document. We calculate user proportion stopping at different vertical positions in a document, as shown in Figure 4 (b). It can be observed that only less than 20% users early stop reading before the end of a document (before 90% vertical position of a document) and most users tend to read almost the full document. It seems to violate the assumption in many related works [18, 34]. Thus, we also attempt to apply this heuristic into retrieval models and study its effectiveness in information retrieval.

3.7 Reviewing existing retrieval models

We review existing retrieval models with the summarized reading heuristics. In Table 2, it can be observed that most retrieval models only satisfy a few reading heuristics. For representation-based models, query and document are mapped into semantic embedding. They thus lose a lot of fine-grained information and do not satisfy any reading heuristics in Table 1. Due to their simplicity, representation-based models often cannot perform as well as interaction-based models in most retrieval scenarios [17, 20].

Interaction-based models calculate the local interactions of each query and document at input and learn term-level interaction patterns for estimating relevance. They thus satisfy the reading heuristic of *Query Centric Guidance* since they utilize exact query match or semantic query match approach. However, it is found that most interaction-based models only satisfy this reading heuristic because they strongly assume each piece of context is independent and only contribute to improving matching problem in different ways. DeepRank [22] simulates the human judgement process and models interaction selectively on query centric context only, which satisfies the reading heuristic of *Selective Attention*. But its selective attention is fixed and only modeled on query centric context. As for HiNT [6], it sequentially models passage-level information and accumulates to final relevance. The sequential modeling and accumulative decision strategy make it satisfy the reading heuristics

Table 2: Comparison between different retrieval models with reading heuristics.

Type	Model	a	b	c	d	e	f
Representation	ARC-I [11]						
Interaction	ARC-II [11]			✓			
	DRMM [9]			✓			
	MatchPyramid [21]			✓			
	KNRM [32]			✓			
	PACRR [13]			✓			
	DeepRank [22]			✓		✓	
	HiNT [6]	✓		✓	✓		

of *Sequential Reading* and *Context-aware Reading*. Since the k-max pooling layer in HiNT selects top-k signals over all the positions, which are based on all the passages in a document, it can not be considered as a reading heuristic of *Selective Attention.*

We can observe that most retrieval models only satisfy a few reading heuristics in Table 1. Therefore, we implement a new retrieval model to consider the reading heuristics that are not applied in existing retrieval models, as detailed in Section 4.4.

4 READING INSPIRED MODEL

In this section, we introduce a Reading Inspired Model (RIM) which can incorporate all the reading heuristics in Section 3. We first introduce our model and then discuss how to model the proposed reading heuristics.

4.1 Model Overview

Given an input document d with T sentences and a query q, our model aims to estimate the relevance of this document. We categorize the reading heuristics into implicit (a-d) and explicit (e-f) types. For implicit reading heuristic, we design specific components in our model for these heuristics. In particular, the heuristic of *Query centric guidance* instructs how to model the semantic matching between query and document, which is detailed in Section 4.2. As for the explicit reading heuristics, we model them as actions in a Markov Decision Process and the decision sequence is optimized by reinforcement learning. In our work, we consider sentence as the atomic reading unit. *Selective attention* is modeled as an action to decide whether to read or skip each input sentence. *Early stop reading* is modeled as an action to decide when to stop reading when the collected information is convincing enough to estimate the document relevance. Figure 5 shows a general schema of our proposed Reading Inspired Model (RIM). It is trained by reinforcement learning with a defined reward based on the performance of relevance estimation.

4.2 Local Matching Layer

The local matching layer aims to capture the semantic matching between query and sentence, which is instructed by the heuristic of *Query centric guidance.* The basic idea is to follow IR heuristics [7, 17] and qualify them into retrieval models. According to previous work, such heuristics include exact query matching and semantic query matching [7, 17], proximity [29] and term importance [17].

Following the idea in [6], we apply term-level interaction matrix with both exact query matching and semantic query matching. The architecture of our local matching layer is shown in Figure 5. Specifically, for a given query $\mathbf{q} = [w_1, w_2, ..., w_m]$ and a document $\mathbf{d}$ with T sentences, where each sentence is $\mathbf{s} = [v_1, v_2, ..., v_n]$, we construct a semantic matching matrix M^{cos} and an exact matching matrix M^{xor}, which are defined as follows:

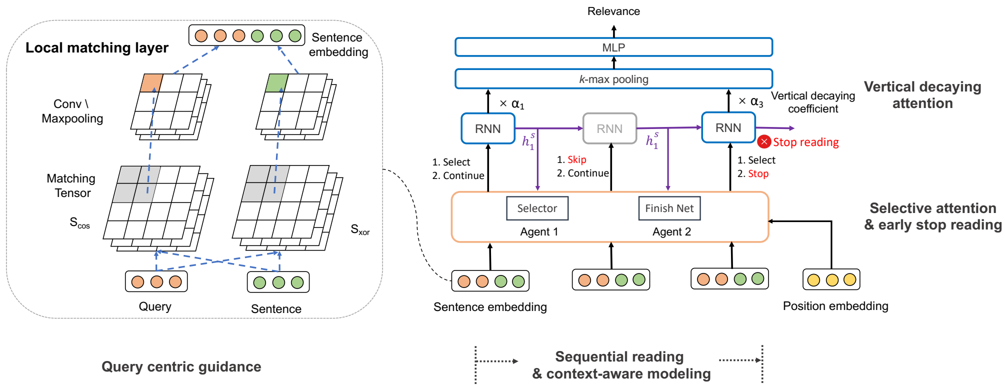


Figure 5: General schema of the Reading Inspired Model (RIM).

$$M_{ij}^{cos} = cos(w_i, v_j), \tag{1}$$

$$M_{ij}^{xor} = \begin{cases} 1, & w_i = v_j \\ 0, & otherwise \end{cases} \tag{2}$$

Exact matching and semantic matching provide critical signals for information retrieval as suggested by [7, 17]. To further incorporate term importance to the input, we extend each element M_{ij} to a three-dimensional representation vector $S_{ij} = [x_i, y_j, M_{ij}]$ by concatenating two term embeddings as in [6], where $x_i = w_i * \boldsymbol{W_c}$ and $y_j = v_j * \boldsymbol{W_c}$. $\boldsymbol{W_c}$ is a compressed matrix to be learned during training. The proximity of each word matching is retained in these matching matrices.

Based on two interaction matrices, we further apply CNN to generate local relevance embedding, which is also called sentence embedding. Note that CNN is more efficient than spatial GRU applied in [6] and it can also capture the relation between several adjacent words. The final sentence embedding is represented by concatenating the signals from two interaction matrices:

$$\boldsymbol{s} = [CNN(S^{cos}), CNN(S^{xor})] \tag{3}$$

4.3 Implicit heuristic modeling

Implicit reading heuristics (*a*-*d*) are modeled in the designed components of model framework. The general schema of our proposed model is shown in Figure 5. Based on the sentence embedding from local matching layer, our model sequentially processes each input sentence, which follows the reading heuristic of *Sequential reading*. Whether a sentence is read is decided by the selector of RIM as a sequential decision process. If the sentence is selected, it is transferred to a RNN module:

$$\boldsymbol{h}_t^s = RNN(\boldsymbol{h}_{t-1}^s, \boldsymbol{s}_t), t = 1, ..., T' \tag{4}$$

where T' is the number of total selected sentences before stopping reading ($T' \leq T$). Modeling sentences by RNN module is able to capture the context information in neighboring sentences, which follows the reading heuristic of *Context-aware reading*. It can be *Context independent* if we replace the RNN module with a simple non-linear layer.

We define a decaying coefficient to model the reading heuristic of *Vertical decaying attention*, which is based on users' reading distribution in real reading behavior data [16]. Specifically, we utilize a Gamma distribution to fit users' fixation distribution in each vertical position, as the blue line in Figure 2:

$$\alpha(p) = \frac{(p-l)^{k-1}}{\Gamma(k)\theta^k} exp(-\frac{p-l}{\theta}) \tag{5}$$

where p is the vertical position in a document, l, k, θ is the location parameter, shape parameter and scale parameter, respectively. After fitting the data, we have $l = 1.36, k = 4.37$ and $\theta = 1.36$. The obtained decaying coefficient is used to multiply the output selected sentence hidden state $\boldsymbol{h}_t^s$:

$$\boldsymbol{h}_t^v = \boldsymbol{h}_t^s * \alpha(p_t), t = 1, ..., T' \tag{6}$$

The hidden state $\boldsymbol{h}_{1:T'}^v$ are then utilized to estimate relevance by a k-max pooling layer and a full connected layer. k-max pooling layer selects top-k signals over all the selected sentences and full connected layer maps hidden states to a relevance score.

4.4 Explicit heuristic modeling with Reinforcement Learning

Selective attention and *Early stop reading* (*e*-*f*) are considered as explicit reading heuristics, which are modeled as a Markov Decision Process. In Figure 5, an agent *Selector* controls our model whether to read input sentence or consider it as irrelevant information and skip it. The other agent *Finish Net* decides whether the collected information is enough to stop reading and estimate a document relevance. The decision policies of two agents are:

$$\begin{aligned} \pi(a_t^s|\boldsymbol{s}_t, \boldsymbol{h}_{t-1}^s, p_t) &= \sigma(\boldsymbol{W_s} * [\boldsymbol{s}_t, \boldsymbol{h}_{t-1}^s, posEmb(p_t)] + b_s) \\ \pi(a_t^f|\boldsymbol{s}_t, \boldsymbol{h}_{t-1}^s, p_t) &= \sigma(\boldsymbol{W_f} * [\boldsymbol{s}_t, \boldsymbol{h}_{t-1}^s, posEmb(p_t)] + b_f) \end{aligned} \tag{7}$$

where σ is the *sigmoid* function, the state at each step is the concatenation of sentence embedding $\boldsymbol{s}_t$, the hidden state of previous selected sentences $\boldsymbol{h}_{t-1}^s$ and position embedding of p_t. *posEmb* maps the position p at step t into an embedding vector. During training, the action is sampled according to the probability in Equation 7. In testing, the action with maximal probability (i.e., $a_t^* = \arg\max \pi(a_t|\theta_t)$) will be selected for superior prediction.

Our model uses a delayed reward to guide the policy learning. Once all the actions are sampled by our model, the representation of the document is determined and passed to estimate relevance. The performance of relevance prediction is considered as a feedback to evaluate the generated representation and the sampled actions. We have three different reward types:

$$R = \begin{cases} -\sum_i^K MSE(y_i, \tilde{y_i}), & pointwise \\ -\sum_{d^+}\sum_{d^-} max(0, 1 - y_{d^+} + y_{d^-}), & pairwise \\ NDCG(\boldsymbol{y}_{1:K}, \tilde{\boldsymbol{y}}_{1:K}), & listwise \end{cases} \tag{8}$$

Table 3: Statistics of the dataset in our experiments. *Click* **means the click relevance label from click model while** *Manual* **means human annotated label.**

	QCL-Train	QCL-Test	NTCIR-13	NTCIR-14
# query	534,655	2,000	100	79
# doc	7,682,872	50,150	9985	4816
# doc per query	14.37	25.08	99.85	60.96
Vocabulary Size	821,768	445,885	211,957	135,073
Label type	Click	Click	Manual	Manual

where K is the document number of a result list, y is the predicted relevance score and $\bar{y}$ is the true relevance score. Pairwise reward is based on a pair of positive and negative samples d^+ and d^-. Listwise reward is the list-level evaluation measure such as NDCG.

The parameter of our model is optimized by REINFORCE algorithm [30] and policy gradient methods [28], aiming to maximize the expected reward. The gradient of the policy is given by

$$\begin{aligned}\nabla \mathcal{J}_\Theta(\Theta) &= \mathbb{E}_{\pi_\Theta}\left[\sum_{i=1}^{K}\sum_{t=1}^{T'} \nabla(\log \pi(a^s_{i,t}|\Theta) + \log \pi(a^f_{i,t}|\Theta)) \cdot R\right] \\ &\approx \frac{1}{M}\sum_{m=1}^{M}\sum_{i=1}^{K}\sum_{t=1}^{T'} \nabla(\log \pi(a^s_{(m,i),t}|\Theta) + \log \pi(a^f_{(m,i),t}|\Theta)) \cdot R_m\end{aligned} \tag{9}$$

where Θ denotes all the model parameters, M is the sampled number, K is the document number of a result list, T' is the total number of selected sentences until the model samples a stop reading signal from finish net, i.e., $a^f_{i,t} = 0$ (continue reading) for $t < T'$ and $a^f_{i,T'} = 1$ (stop reading).

Considering that the length of sampled sequences in each document is long and different significantly, the space of policy exploration is often very large, resulting in a large variance of gradient estimation. To reduce the variance, we subtract the reward by a baseline term (the average value of M samples) and truncate negative rewards as in [18]. In addition, to have a balance between exploration and exploitation, a small probability ϵ is set to uniformly sample from the entire action space, as in [18].

The optimization objective in Equation 9 can be considered as an Actor-Critic algorithm [15], where $\pi(a_t|\Theta)$ is the actor and the network for predicting relevance is the critic. Thus, Equation 9 is to optimize the parameters of the actor. Policy gradient method can only backpropagate reward signals to the parameters before the policy network. To optimize the parameter of the critic, we directly optimize MSE between the predicted relevance score and the true relevance score. The final objective of our model is to optimize actor and critic simultaneously.

5 EXPERIMENT

After introducing the reading heuristics and how to incorporate them into retrieval models, in this section, we conduct a series of experiments to investigate the effectiveness of the reading heuristics as well as the retrieval performance of proposed RIM model. In particular, we aim to address

- **RQ1:** Which reading heuristics have positive impacts on the retrieval performance?
- **RQ2:** How does our model RIM perform compared to existing retrieval models when integrating all effective heuristics?
- **RQ3:** Can RIM capture users' reading patterns in explicit decision sequences?

5.1 Dataset

To evaluate the performance of different retrieval models, we conduct experiments on a large-scale public available benchmark data (QCL) [36] and two released test sets from NTCIR 13-14 WWW tasks [14, 19]. Table 3 shows the statistics of the datasets. QCL is sampled from the query log of a popular Chinese commercial search engine. It contains weak relevance labels derived by five different click models for over 12 million query-document pairs. Prior works [2, 32] have shown that such weak relevance labels derived by click models can be used to train and evaluate retrieval models. Thus, in our work, we utilize click relevance label to train our model. The click relevance labels of QCL are derived from five click models, TACM, PSCM, UBM, DBN, and TCM respectively. We use relevance inferred by PSCM to train the retrieval models because the PSCM has the best relevance estimation performance among these five alternatives. Similar to the evaluation settings used in [32], we utilize two different click relevance labels to evaluate our model on the test set of QCL. In the Test-SAME setting, we uses click relevance labels from the same PSCM to evaluate our model. In the Test-DIFF setting, we use UBM [4] as the relevance labels for evaluation. .

In addition, to evaluate the performance of our model on human annotated labels, we utilize the test sets from NTCIR 13-14 WWW tasks (Chinese) [14, 19]. All the query-document pairs in NTCIR WWW tasks are rated by human assessors on a four-point scale following the standard TREC criterion. A drawback of NTCIR WWW datasets is that their size is much smaller than QCL, which limits the statistical power of the evaluation experiments on them. Evaluation on the Test-DIFF setting and NTCIR WWW datasets can measure the generalization ability of retrieval models, because the training labels and testing labels are generated differently [32].

5.2 Experimental settings

We implement all the retrieval models by using Pytorch. The parameters are optimized by Adadelta, with a batch size of 80 and a learning rate of 0.1. The dimension of the embedding layer is 50 and it is initialized with the word2vec trained on a Chinese Wikipedia dataset[1]. The dimension of the hidden vectors is 128 and the dimension of the position embedding is 3. The CNN uses filter windows with sizes 2 to 5 and 64 feature maps for each filter. The RNN we used is Gated Recurrent Unit (GRU). Early stopping with a patience of 10 epochs is adopted during the training process.

We adopt the pointwise reward in the training process because in our experiment it has a better retrieval performance than the pairwise and listwise rewards. For each document, we sample $M = 5$ possible action sequences. The exploration rate ϵ is 0.2. In addition, the number of candidate documents of each query in NTCIR dataset is much more than that of QCL, so we only rerank the top 40 documents with highest BM25 scores.

We compare RIM with the baselines discussed in Section 3.7. In addition, we also implement a **BaseReader**, which removes the explicit reading heuristics and *Vertical decaying attention* in RIM. BaseReader with only *Selective attention* heuristic and *Early stop reading* are called **RIM-select** and **RIM-stop**, respectively. We compare these three models to further investigate the effectiveness of the corresponding reading heuristic in the next section.

5.3 Effectiveness of reading heuristic

In this section, we test the effectiveness of each reading heuristic in the design of retrieval models, which aims to answer **RQ1**. We

[1]http://download.wikipedia.com/zhwiki

Table 4: Ranking Performance of HiNT and BaseReader in different reading order. * indicates the statistical significant improvements with respect to *Sequential reading* (p-value ≤ 0.05).

	Test-SAME (PSCM)				
	Direction	NDCG@1	NDCG@3	NDCG@5	NDCG@10
HiNT	Sequential	**0.7550**	**0.7592**	**0.7751**	**0.8264**
	Inverse	0.6757*	0.6952*	0.7204*	0.7831*
	Random	0.7297*	0.7415*	0.7618	0.8177*
Base reader	Sequential	**0.6988**	**0.7198**	**0.7418**	**0.8008**
	Inverse	0.6803*	0.7091*	0.7328*	0.7941
	Random	0.6799*	0.7099*	0.7355*	0.7981
	NTCIR-13				
	Direction	NDCG@1	NDCG@3	NDCG@5	NDCG@10
HiNT	Sequential	**0.6566**	**0.6599**	**0.6548**	0.6449
	Inverse	0.6533	0.6466	0.6550	**0.6531**
	Random	0.5983	0.6041	0.6008	0.6058
Base reader	Sequential	**0.6566**	**0.6559**	**0.6546**	**0.6482**
	Inverse	0.6400	0.6512	0.6475	0.6396
	Random	0.6200	0.6304	0.6333	0.6314

Table 5: Ranking Performance of HiNT and BaseReader when adding vertical decaying coefficient. * indicates the statistical significant improvements with respect to the original performance (p-value ≤ 0.05).

	Test-SAME (PSCM)				
		NDCG@1	NDCG@3	NDCG@5	NDCG@10
HiNT	Original	0.7550	0.7592	0.7751	**0.8264**
	Add $\alpha(p)$	**0.7607**	**0.7616**	**0.7778**	0.8177*
Base reader	Original	**0.6988**	**0.7198**	**0.7418**	**0.8008**
	Add $\alpha(p)$	0.6869	0.7064	0.7299*	0.7915
	NTCIR-13				
		NDCG@1	NDCG@3	NDCG@5	NDCG@10
HiNT	Original	**0.6566**	**0.6599**	**0.6548**	**0.6449**
	Add $\alpha(p)$	0.6217	0.6380	0.6423	0.6439
Base reader	Original	0.6566	0.6559	0.6546	**0.6482**
	Add $\alpha(p)$	**0.6567**	**0.6584**	**0.6581**	0.6453

utilize these HiNT and the BaseReader in RIM to evaluate each reading heuristics. In our experiment, we do not use ablation study to analyze the effectiveness of *Query centric guidance*, because it has been demonstrated in prior studies [7, 17]. In addition, due to the space limitations, we only report the results on Test-SAME and NTCIR-13 because results on the other datasets are similar.

5.3.1 Sequential reading. This reading heuristic indicates that the reading direction of retrieval models which is generally from top to bottom, will influence their performance.. To test whether this implication holds, we change the order of the sentences in document to *inverse* and *random*. The results are shown in Table 4.

We observe that two models applying *Sequential reading* achieve best performance in both datasets. In Test-SAME, the improvement over *inverse* and *random* is significant. Due to the size of NTCIR dataset, the difference between *Sequential reading* and other reading directions is not significant but *Sequential reading* still performs best. This illustrates that reading direction is important for retrieval models and the heuristic of *Sequential reading* can help improve ranking performance.

5.3.2 Vertical decaying attention. This heuristic comes from the findings that users' reading attention is gradually decaying in the vertical direction [16]. We adopt Gamma distribution to fit users' fixation distribution during relevance judgement in the user study data [16] and use it as a decaying factor for the output sentence embedding in the HiNT and BaseReader. The result is shown in Table 5.

It is observed that in both datasets, adding the decaying coefficients does not change the retrieval performance of these two

Table 6: Ranking Performance of HiNT and BaseReader in context-aware reading and context-independent reading. * indicates the statistical significant improvements with respect to the context-aware reading (p-value ≤ 0.05).

	Test-SAME (PSCM)				
		NDCG@1	NDCG@3	NDCG@5	NDCG@10
HiNT	context-aware	**0.7550**	**0.7592**	**0.7751**	**0.8264**
	independent	0.6731*	0.6933*	0.7183*	0.7814*
Base reader	context-aware	**0.6988**	**0.7198**	**0.7418**	**0.8008**
	independent	0.6650*	0.6684*	0.6823*	0.7693*
	NTCIR-13				
		NDCG@1	NDCG@3	NDCG@5	NDCG@10
HiNT	context-aware	0.6566	0.6599	**0.6548**	0.6449
	independent	**0.6633**	**0.6600**	0.6483	**0.6461**
Base reader	context-aware	**0.6566**	**0.6559**	**0.6546**	**0.6482**
	independent	0.6517	0.6475	0.6425	0.6478

Table 7: Ranking Performance of BaseReader when applying explicit reading heuristics. * indicates the statistical significant improvements with respect to BaseReader (p-value ≤ 0.05).

	Test-SAME (PSCM)			
	NDCG@1	NDCG@3	NDCG@5	NDCG@10
BaseReader	0.6988	0.7198	0.7418	0.8008
RIM-select	0.7359*	0.7604*	0.7609*	0.8130*
RIM-stop	**0.7634***	**0.7637***	**0.7776***	**0.8265***
	NTCIR-13			
	NDCG@1	NDCG@3	NDCG@5	NDCG@10
BaseReader	0.6566	0.6559	0.6546	0.6482
RIM-select	0.6733	0.6646	0.6656	**0.6560**
RIM-stop	**0.6917**	**0.6752**	**0.6607**	0.6431

models significantly. The results suggest that it may be not suitable to incorporate this heuristic directly by adding a decaying coefficient. While this experiment failed to prove the effectiveness of this heuristic, we leave the investigation of how to properly incorporate it into retrieval models for future work.

5.3.3 Context-aware reading. This heuristic comes from the findings that users' reading behavior is context-dependent. Given a specific reading direction, the output sentence embedding is influenced by the previously read text. We adopt the different design as in Figure 3 to show the effectiveness of *Context-aware reading* in retrieval models. The result is shown in Table 6.

We can observe that the performances over these two datasets are different. In Test-SAME, the models associated with *Context-aware reading* is significantly better than those with *Context-independent reading*. However, when testing on the human annotated labels in NTCIR-13, the differences between two heuristics are very small. Similar results are also observed in Test-DIFF and NTCIR-14 setting. The results reveal the gap when applying retrieval models in different test settings. The heuristic of *Context-aware reading* can help improve the performance in terms of click relevance labels but not in terms of human annotated labels.

5.3.4 Explicit reading heuristic. Explicit reading heuristics are modeled as actions in Markov Decision Process and learned by using reinforcement learning. In our work, we only extend BaseReader with the proposed action policies and compare it with the original BaseReader. RIM-select and RIM-stop incorporate the reading heuristic of *Selective attention* and *Early stop reading*, respectively. The result is shown in Table 7.

We observe that when applying these two explicit reading heuristics, our models achieve better performance in most of evaluation metrics compared to the original BaseReader. The improvement is also significant in Test-SAME, which implies that these two

Table 8: Ranking performance of different retrieval models over QCL test set, NTCIR-13 and NTCIR-14. * indicates the statistical significant improvements with respect to RIM (p-value ≤ 0.05).

	Test-SAME (PSCM)				Test-DIFF (UBM)			
	NDCG@1	NDCG@3	NDCG@5	NDCG@10	NDCG@1	NDCG@3	NDCG@5	NDCG@10
BM25	0.7048*	0.7202*	0.7414*	0.7967*	0.6127*	0.6509*	0.6819*	0.7429*
ARC-I	0.7583*	0.7647	0.7804	0.8286	0.6489*	0.6869	0.7142	0.7677
ARC-II	0.7239*	0.7347*	0.7519*	0.8061*	0.6303*	0.6667*	0.6948*	0.7523*
DRMM	0.6958*	0.7141*	0.7352*	0.7923*	0.6024*	0.6471*	0.6790*	0.7404*
MatchPyramid	0.6851*	0.7028*	0.7248*	0.7857*	0.5938	0.6386	0.6716*	0.7366*
KNRM	0.6997*	0.7121*	0.7336*	0.7917*	0.6048	0.6465*	0.6775*	0.7400*
PACRR	0.7072*	0.7219*	0.7411*	0.7981*	0.6172*	0.6557*	0.6860*	0.7465*
DeepRank	0.7058*	0.7227*	0.7452*	0.8059*	0.6099*	0.6566*	0.6891*	0.7540*
HiNT	0.7550*	0.7592*	0.7751*	0.8264	0.6564	0.6895	0.7072*	0.7603*
RIM	**0.7746**	**0.7705**	**0.7830**	**0.8304**	**0.6602**	**0.6918**	**0.7170**	**0.7689**

	NTCIR-13				NTCIR-14			
	NDCG@1	NDCG@3	NDCG@5	NDCG@10	NDCG@1	NDCG@3	NDCG@5	NDCG@10
BM25	0.6099	0.6194	0.6253	0.6391	0.4324	0.4432	0.4383	0.4706
ARC-I	0.5933	0.6153	0.6184	0.6222	0.4726	0.4690	0.4643	0.4814
ARC-II	0.6466	0.6649	0.6523	0.6426	0.4556	0.4369	0.4405	0.4700
DRMM	0.6866	0.6490	0.6487	0.6378	0.4345	0.4651	0.4657	0.4847
MatchPyramid	0.6866	0.6507	0.6458	0.6436	0.3586	0.3838	0.3998	0.4378
KNRM	0.6700	0.6564	0.6557	0.6591	0.4367	0.4252	0.446	0.4739
PACRR	0.6700	0.6661	0.6659	0.6620	0.4219	0.4483	0.4541	0.4689
DeepRank	0.6750	0.6606	0.6617	**0.6648**	0.4894	0.4588	0.4640	0.4793
HiNT	0.6566	0.6599	0.6548	0.6449	0.4746	0.4643	0.4617	0.4898
RIM	**0.7050**	**0.6797**	**0.6749**	0.6570	**0.4979**	**0.4887**	**0.4911**	**0.5021**

reading heuristics are effective in improving retrieval models. RIM-select is able to find the key supporting sentences for relevance modeling and RIM-stop reduces redundancy information fed to the BaseReader when the previously read text is enough to judge relevance.

Recalling the IR heuristics proposed in [7], the RIM-select and RIM-stop can skip irrelevant information in a document, and therefore reduce the document length. If the skipped text do not contain query terms, the improvement of our models can be simply explained by the Length Normalization Constraints [7]. However, according to our statistic, the read sentences only cover 53% and 41% of total query terms in a document by RIM-select and RIM-stop, respectively. This illustrates that neural retrieval models may not performs worse when removing query centric context. Instead, if the retained sentences are the key supporting ones, the performance will even be better.

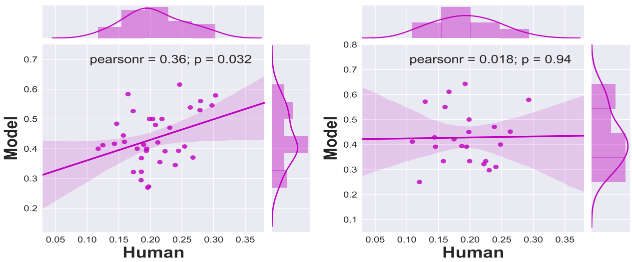
5.3.5 Summary. For **RQ1**, based on the experimental results, we can find that most of reading heuristics have positive impacts on the retrieval performance. However, modeling the heuristic of *Vertical decaying attention* by adding a vertical decaying coefficient does not bring improvement for the retrieval models. *Context-aware reading* only has positive impacts on retrieval performance when testing on homogeneous click relevance label, but its effectiveness is not validated on human annotated labels. In this case, we also consider *Context-aware reading* as a potentially effective heuristic. In summary, except for the *Vertical decaying attention* heuristics, all the reading heuristics help to improve the retrieval performance of the HiNT and BaseReader. Therefore, we integrate all of them except *Vertical decaying attention* into RIM.

5.4 Overall performance

In this section, we aim to address **RQ2**. We show that five of the six reading heuristics are effective in the previous section. So we integrate them into the RIM. We compare it with existing retrieval models over the QCL test set and NTCIR 13-14 test sets. The results are summarized in Table 8.

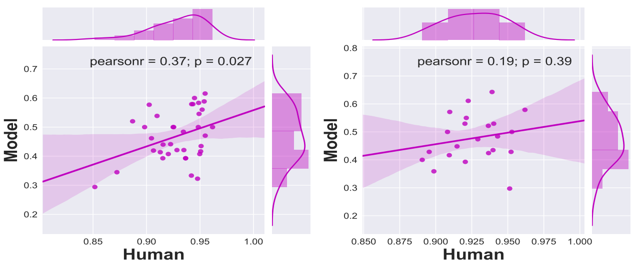
It is observed that different retrieval models perform differently on two kinds of datasets. On the QCL test set, we can see that BM25 is a strong baseline which outperforms most retrieval models. This illustrates the performance of BM25 is close to most retrieval models if testing on click relevance labels. ARC-I, as a representation-based model, outperforms all the baselines in Test-SAME and Test-DIFF, which indicates its effectiveness on click relevance labels. For the models with only one reading heuristic (ARCI-II, DRMM, MatchPyramid, KNRM, PACRR), we can find that they perform similarly in Test-SAME and Test-DIFF. DeepRank inherits the reading heuristic of *Selective attention* but this selective attention is fixed on the query centric context only. We can observe that DeepRank performs similarly with most retrieval models, which illustrates fixed selective attention does not significantly improve ranking performance. The HiNT that models more reading heuristics, outperforms most baselines in different metrics. The RIM, it incorporates all the effective reading heuristics and outperforms other retrieval models, which again demonstrates that the reading heuristics is important for designing a better retrieval model.

As for human annotated labels, we find that the results are different from those based on click relevance labels, showing a gap between different test settings. It is observed that BM25 only outperforms the representation-based model ARC-I. DRMM and MatchPyramid have good performance on $NDCG@1$, but the performance on $NDCG@10$ is not as good as other models like KNRM. KNRM and PACRR, although following only the heuristic of *query centric guidance*, have relatively good performance on four different metrics since they extend simple query matching to different soft-level matching. We can also see that the RIM outperforms other baselines on most evaluation metrics, which illustrates that although our model is trained based on click relevance labels, it can still perform well on human annotated labels.



(a) Relevant documents **(b) Irrelevant documents**

Figure 6: Selected ratio of human behavior and model decision.



(a) Relevant documents **(b) Irrelevant documents**

Figure 7: Stop reading position of human behavior and model decision.

5.5 Comparison with users' reading behavior

This section aims to answer **RQ3**. To compare our model's decision sequences on explicit reading heuristics with real user's behavior data, we test RIM on a public reading behavior dataset [16], which contains 15 queries and 60 documents in total. The number of relevant and irrelevant documents are 37 and 23, respectively. For the heuristic of *Selective attention*, we calculate the average ratio of texts read by users and compare it with the ratio of texts selected by RIM. Similarly, users' stop reading position is compared with the stop positions of RIM for the heuristic of *Early stop reading*. We compare the differences between relevant and irrelevant documents. The results are shown in Figures 6 and 7.

We can observe that for relevant documents, the selected ratio and stop reading position of RIM have a significant linear relationship with users' behaviors, where correlations are 0.36 and 0.37, respectively (p-value$<$0.05). It illustrates that RIM can capture real users' reading patterns to some extent. However, for irrelevant documents, the selected ratio and stop reading position of RIM are both weakly related to users' behaviors. It is probably due to the fact that user behavior in irrelevant documents is more uncertain, which increases the learning complexity of our model.

Although RIM can capture related users' reading patterns, from the marginal distributions in Figures 6 and 7, we observe that the selected ratio and stop reading position between human and our model are different. In Figure 6, RIM selects about 43% sentences while users only read about 20% texts in a document. This difference is due to that human have parafoveal preview during the reading process [23]. Users read only a few texts but actually obtain more information based on the reading ratio calculated by eye fixations (i.e., more than 20% contents). However, retrieval models do not have this biological mechanism, thus need to read more texts to judge relevance than human. From the marginal distribution in Figure 7, we find that users generally stop at more than 90% vertical position of a document but RIM stops at about 47% vertical position. This illustrates that early stop reading is not common in real users but RIM tends to stop earlier than users' behaviors. We further study why users tend to read almost the full document. According to our statistics, the average ratio of users' reading texts after 70% of a document is only 15%, which illustrates that users' main reading attention is not located in the bottom of a document. Users are more likely to quickly scan these contents to reexamine their relevance judgement based on previously read texts, which is also observed in [16]. Our model decides to directly skip these contents since the previously read texts are enough to make relevance judgment.

For both heuristics, we can find that although RIM can capture a similar reading pattern as user's behavior in relevant documents, the selected ratio and stop reading position between human and model are still different. The reason may be that users have a particular biological mechanism, which cannot be simulated by retrieval models. However, retrieval models utilize their specific strategies to remedy this difference and estimate reasonable relevance, such as selecting more texts to read and directly skipping the rest unimportant texts. In its essence, reading heuristics can indeed help retrieval models to improve retrieval performance. But retrieval models may perform these heuristics in a different way compared to users' reading behavior.

6 CONCLUSION

In this paper, we investigate users' reading patterns during relevance judgement and propose six reading heuristics. It is observed that a large number of popular retrieval model only satisfy a part of these reading heuristics. These heuristics are further incorporated into a newly proposed Reading Inspired Model (RIM) in different ways, where implicit heuristics are directly incorporated into the model framework and explicit heuristics are learned with a reinforcement learning method. By conducting an ablation study, we show that most reading heuristics have positive impacts on retrieval performance. As for the heuristic of *Vertical decaying attention*, we find that directly adding a decaying coefficient is not effective for improving retrieval performance. In addition, although the heuristic of *Context-aware reading* is found only effective on homogeneous click relevance label (QCL), we also consider it as an effective heuristic for retrieval models. In short, we integrate all the heuristics except *Vertical decaying attention* into our proposed model RIM. Experimental results on a large-scale benchmark dataset QCL and NTCIR WWW test sets demonstrate that RIM outperforms all the baselines in terms of different evaluation metrics. In addition, we compare real users' reading patterns with the explicit decision sequences of our model. We observe that our model can indeed capture similar reading patterns as user behavior. Our work provides deeper sights into the reading heuristics on retrieval models and improves both retrieval performance and explainability.

In the future, we plan to further study the heuristic of *Query centric guidance* and *Vertical decaying attention*. Specifically, *Query centric guidance* plays a pivot role in retrieval models and different matching strategies can bring significant different retrieval performances. We also plan to study other approaches to model *Vertical decaying attention* instead of directly adding a vertical decaying coefficient. We believe that a deeper understanding of these two heuristics can further help improve the retrieval performance.

7 ACKNOWLEDGEMENTS

This work is supported by Natural Science Foundation of China (Grant No. 61622208, 61732008, 61532011) and the National Key Research and Development Program of China (2018YFC0831700).

REFERENCES

[1] Michael Bendersky, Donald Metzler, and W Bruce Croft. 2010. Learning concept importance using a weighted dependence model. (2010), 31–40.

[2] A Chuklin, I Markov, and M de Rijke. 2015. Click models for web search. Synthesis lectures on information concepts, retrieval, and services.

[3] Josipa Crnic. 1983. *Introduction to Modern Information Retrieval.*

[4] Georges E Dupret and Benjamin Piwowarski. 2008. A user browsing model to predict search engine click data from past observations.. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, 331–338.

[5] Michael Eisenberg and Carol Barry. 1988. Order effects: A study of the possible influence of presentation order on user judgments of document relevance. *Journal of the American Society for Information Science* 39, 5 (1988), 293–300.

[6] Yixing Fan, Jiafeng Guo, Yanyan Lan, Jun Xu, Chengxiang Zhai, and Xueqi Cheng. 2018. Modeling Diverse Relevance Patterns in Ad-hoc Retrieval. *International ACM SIGIR Conference on Research and development in Information Retrieval* (2018), 375–384.

[7] Hui Fang, Tao Tao, and ChengXiang Zhai. 2004. A formal study of information retrieval heuristics. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, 49–56.

[8] Tsujui Fu and Weiyun Ma. 2018. Speed Reading: Learning to Read ForBackward via Shuttle. (2018), 4439–4448.

[9] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W. Bruce Croft. 2016. A Deep Relevance Matching Model for Ad-hoc Retrieval. In *Acm International on Conference on Information and Knowledge Management.*

[10] Michael Hahn and Frank Keller. 2016. Modeling Human Reading with Neural Attention. *empirical methods in natural language processing* (2016), 85–95.

[11] Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2014. Convolutional Neural Network Architectures for Matching Natural Language Sentences. *Neural Information Processing Systems* (2014), 2042–2050.

[12] Po Sen Huang, Xiaodong He, Jianfeng Gao, Deng Li, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Acm International Conference on Conference on Information and Knowledge Management.*

[13] Kai Hui, Andrew Yates, Klaus Berberich, and Gerard de Melo. 2017. Pacrr: A position-aware neural ir model for relevance matching. *arXiv preprint arXiv:1704.03940* (2017).

[14] Mao Jiaxin, Sakai Tetsuya, Luo Cheng, Xiao Peng, Liu Yiqun, and Dou Zhicheng. 2018. Overview of the ntcir-14 we want web task. *Proc. NTCIR-14, To appear* (2018).

[15] Vijay R Konda and John N Tsitsiklis. 2000. Actor-critic algorithms. In *Advances in neural information processing systems.* 1008–1014.

[16] Xiangsheng Li, Yiqun Liu, Jiaxin Mao, Zexue He, Min Zhang, and Shaoping Ma. 2018. Understanding Reading Attention Distribution during Relevance Judgement. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management.* ACM, 733–742.

[17] Pang Liang, Yanyan Lan, Jiafeng Guo, Jun Xu, and Xueqi Cheng. 2017. A Deep Investigation of Deep IR Models. (2017).

[18] Xianggen Liu, Lili Mou, Haotian Cui, Zhengdong Lu, and Sen Song. 2018. Jumper: Learning when to make classification decisions in reading. *arXiv preprint arXiv:1807.02314* (2018).

[19] Cheng Luo, Tetsuya Sakai, Yiqun Liu, Zhicheng Dou, Chenyan Xiong, and Jingfang Xu. 2017. Overview of the ntcir-13 we want web task. *Proc. NTCIR-13* (2017).

[20] Yifan Nie, Yanling Li, and Jian-Yun Nie. 2018. Empirical Study of Multi-level Convolution Models for IR Based on Representations and Interactions. In *Proceedings of the 2018 ACM SIGIR International Conference on Theory of Information Retrieval.* ACM, 59–66.

[21] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2016. Text Matching as Image Recognition.. In *AAAI.* 2793–2799.

[22] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Jingfang Xu, and Xueqi Cheng. 2017. Deeprank: A new deep architecture for relevance ranking in information retrieval. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management.* ACM, 257–266.

[23] K Rayner. 2009. Eye movements and attention in reading, scene perception, and visual search. *Quarterly Journal of Experimental Psychology* 62, 8 (2009), 1457.

[24] Erik D Reichle, Alexander Pollatsek, Donald L Fisher, and Keith Rayner. 1998. Toward a model of eye movement control in reading. *Psychological Review* 105, 1 (1998), 125–157.

[25] E. D. Reichle, K Rayner, and A Pollatsek. 2003. The E-Z reader model of eye-movement control in reading: comparisons to other models. *Behavioral and Brain Sciences* 26, 4 (2003), 445–476.

[26] Stephen E Robertson and Steve Walker. 1994. Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. *International ACM SIGIR Conference on Research and development in Information Retrieval* (1994), 232–241.

[27] Yelong Shen, Posen Huang, Jianfeng Gao, and Weizhu Chen. 2017. ReasoNet: Learning to Stop Reading in Machine Comprehension. *knowledge discovery and data mining* (2017), 1047–1055.

[28] Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. 2000. Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems.* 1057–1063.

[29] Tao Tao and ChengXiang Zhai. 2007. An exploration of proximity measures in information retrieval. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, 295–302.

[30] Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning* 8, 3-4 (1992), 229–256.

[31] Ho Chung Wu, Robert WP Luk, Kam-Fai Wong, and KL Kwok. 2007. A retrospective study of a hybrid document-context based retrieval model. *Information processing & management* 43, 5 (2007), 1308–1331.

[32] Chenyan Xiong, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, and Russell Power. 2017. End-to-End Neural Ad-hoc Ranking with Kernel Pooling. *International ACM SIGIR Conference on Research and development in Information Retrieval* (2017).

[33] Adams Wei Yu, Hongrae Lee, and Quoc V Le. 2017. Learning to Skim Text. *meeting of the association for computational linguistics* 1 (2017), 1880–1890.

[34] Keyi Yu, Yang Liu, Alexander G Schwing, and Jian Peng. 2018. Fast and Accurate Text Classification: Skimming, Rereading and Early Stopping. (2018).

[35] Tianyang Zhang, Minlie Huang, and Li Zhao. 2018. Learning Structured Representation for Text Classification via Reinforcement Learning. (2018), 6053–6060.

[36] Yukun Zheng, Zhen Fan, Yiqun Liu, Cheng Luo, Min Zhang, and Shaoping Ma. 2018. Sogou-QCL: A New Dataset with Click Relevance Label. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval.* ACM, 1117–1120.