# Modeling Diverse Relevance Patterns in Ad-hoc Retrieval

Yixing Fan[†,‡], Jiafeng Guo[†,‡], Yanyan Lan[†,‡], Jun Xu[†,‡], Chengxiang Zhai* and Xueqi Cheng[†,‡]

†University of Chinese Academy of Sciences, Beijing, China
‡CAS Key Lab of Network Data Science and Technology, Institute of Computing Technology,
Chinese Academy of Sciences, Beijing, China
*Department of Computer Science University of Illinois at Urbana-Champaign, IL, USA
fanyixing@software.ict.ac.cn,{guojiafeng,lanyanyan,junxu,cxq}@ict.ac.cn,czhai@illinois.edu

## ABSTRACT

Assessing relevance between a query and a document is challenging in ad-hoc retrieval due to its diverse patterns, i.e., a document could be relevant to a query as a whole or partially as long as it provides sufficient information for users' need. Such diverse relevance patterns require an ideal retrieval model to be able to assess relevance in the right granularity adaptively. Unfortunately, most existing retrieval models compute relevance at a single granularity, either document-wide or passage-level, or use fixed combination strategy, restricting their ability in capturing diverse relevance patterns. In this work, we propose a data-driven method to allow relevance signals at different granularities to compete with each other for final relevance assessment. Specifically, we propose a HIerarchical Neural maTching model (HiNT) which consists of two stacked components, namely local matching layer and global decision layer. The local matching layer focuses on producing a set of local relevance signals by modeling the semantic matching between a query and each passage of a document. The global decision layer accumulates local signals into different granularities and allows them to compete with each other to decide the final relevance score. Experimental results demonstrate that our HiNT model outperforms existing state-of-the-art retrieval models significantly on benchmark ad-hoc retrieval datasets.

## CCS CONCEPTS

• **Information systems → Learning to rank**;

## KEYWORDS

relevance patterns; ad-hoc retrieval; neural network

## 1 INTRODUCTION

A central question in ad-hoc retrieval is how to learn a generalizable function that can well assess relevance between a query and a document. One of the major difficulties for relevance assessment lies in that there might be diverse relevance patterns between a query and a document. As revealed by the evaluation policy in TREC ad-hoc task [6, 32], "a document is judged relevant if any piece of it is relevant (regardless of how small the piece is in relation to the rest of the document)[1]". In other words, a document could be relevant to a query as a whole or partially as long as it provides sufficient information for users' needs. Such diverse relevance patterns might be highly related to the heterogeneity of long documents in ad-hoc retrieval. As discussed by Robertson and Walker [30], there are two underlying hypotheses concerning document structures in relevance judgement, i.e. verbosity hypothesis and scope hypothesis [30]. With the *Verbosity Hypothesis* a long document might be relevant to a query as a whole, while with the *Scope Hypothesis* the relevant parts could be in any position of a long document, and thus it could be partially relevant to a query.

The diverse relevance patterns call for a retrieval model to be able to assess relevance at the right granularity adaptively in ad-hoc retrieval. Unfortunately, most existing retrieval models operate at a single granularity, either document-wide or passage-level. Specifically, document-wide approaches compare a document as a whole to a query. For example, most probabilistic retrieval methods (e.g. BM25 or language models) and learning-to-rank models [8, 15] rely on document-wide feature statistics for relevance computation. Obviously, such document-wide approaches are difficult to model finer-granularity relevance signals, leading to potential biases on the competition between long and short documents [25]. On the other hand, Passage-level approaches segment a document into passages and aggregate passage-level signals for relevance computation [4, 20, 31]. However, the performance of existing passage-based approaches is mixed when applied to a variety of test beds [35] by only using simple manually designed operations over the passage-level signals. There have been a few efforts [2, 35] trying to combine both document-wide and passage-level methods. For example, Bendersky et al. [2] integrated the query-similarity on a document and its passages using document-homogeneity. Wang et al. [35] combined the document retrieval results with passage retrieval results using a heuristic function [19]. However, by using a fixed combination strategy, these models cannot fully capture the diverse relevance patterns for different query-document pairs.

---

[1]http://trec.nist.gov/data/reljudge_eng.html

Recently, deep neural models have been applied to ad-hoc retrieval. These data-driven methods have shown their expressive power in end-to-end learning relevance matching patterns between queries and documents [10, 14, 23]. However, most existing neural matching models, either representation-focused [14] or interaction-focused [10], belong to the document-wide approaches. For example, the representation-focused models aim to learn a document representation to compare with the query representation, while the interaction-focused models learn from a matching matrix/histogram between a document and a query. To the best of our knowledge, so far there have been no neural matching model proposed to learn relevance signals from both document-wide and passage-level explicitly for modeling diverse relevance patterns in ad-hoc retrieval.

In this paper, we propose a data-driven method to automatically learn relevance signals at different granularities (i.e. passage-level and document-wide), and allow them to compete with each other for final relevance assessment. Specifically, we propose a HIerarchical Neural maTching model (HiNT) which consists of two stacked components, namely local matching layer and global decision layer. The local matching layer focuses on producing a set of local relevance signals between a query and each passage of a document. Many well-known information retrieval (IR) heuristics that characterize the relevance matching between a query and a passage can be encoded in this layer for high quality signal generation. Specifically, we employ a spatial GRU model [34] for the relevance matching between a query and a passage, which can well capture semantic relations, proximities, and term importance. The global decision layer aims to accumulate passage-level signals into different granularities and allow them to compete with each other to form the final relevance score. Flexible strategies are applied in this layer to model diverse relevance patterns. Specifically, we utilize a hybrid network architecture to accumulate local signals, and select signals from both passage-level and document-wide to generate the final relevance score.

We evaluate the effectiveness of the proposed model based on two representative ad-hoc retrieval benchmark datasets from the LETOR collection [28]. For comparison, we take into account several well-known traditional retrieval models, learning to rank models, and deep neural matching models. These models belong to document-wide, passage-level and hybrid approaches. The empirical results show that our model can outperform all the baselines in terms of all the evaluation metrics. We also provide detailed analysis on HiNT model, and conduct case studies to verify the diverse relevance patterns captured by our model over different query-document pairs.

## 2 RELATED WORK

A large number of retrieval methods have been proposed in the past few decades [4, 23, 28, 30]. Without loss of generality, we divide existing methods into three folds, namely document-wide approaches, passage-level approaches and hybrid approaches, based on what kind of relevance signals they rely on for relevance assessment. We will briefly review these studies in the follows.

### 2.1 Document-wide Approaches

Document-wide approaches, by its name, collect and make relevance assessment based on document-wide signals. There have been a large number of retrieval models under this branch. Firstly, traditional retrieval models [1, 30, 39], collect lexical matching signals (e.g., term frequency) from the whole document, and make relevance assessment under some probabilistic framework. For example, the well-known BM25 model [30] collects term frequencies and document length and employs a scoring function derived under the 2-poisson model to compute relevance based on these document-wide signals. Secondly, most machine learning based retrieval methods, including learning to rank models and deep learning models, are also belong to this branch. For learning to rank methods [3, 15, 38], they typically involve two stages, a feature construction stage and a model learning stage. The feature construction stage could be viewed as to define (or collect) relevance signals for a document given a query. Typically, there are three type of features, including query dependent features, document independent features, and query-document dependent features. Most of these features are defined at document level, such as tf-idf scores, BM25 scores, and PageRank. Based on these features, linear [15] or non-linear [3, 38] models are learned to produce the relevance score by optimizing some ranking based loss functions. For deep learning methods [10, 14, 24, 26], they can be categorized into two types according to their architectures [10], namely representation-focused models and interaction-focused models. The representation-focused models, such as ARCI [13] and DSSM [14], aim to learn a low-dimensional abstractive representation for the whole document, and compare it with the query representation. The interaction-focused models, such as DRMM [10] and Duet [24], learn from a matching matrix/histogram between a document and a query to produce a set of document-wide matching signals for final relevance prediction. Although in interaction-focused models, document-wide signals are usually generated from local signals, there is no competition between document-wide signals and local signals for capturing diverse relevance patterns.

Since document-wide approaches take document as a whole, these methods are often difficult to model fine-granularity relevance signals. Meanwhile, by using document-wide statistics as relevance signals, it often leads to certain bias on the competition between long and short documents [25], since long documents are likely to contain stronger signals on average.

### 2.2 Passage-level Approaches

As opposed to document-wide methods, passage-level methods collect signals from passages to make relevance assessment on the document. Note here we focus on document retrieval using passage-level signals, and will not include the work taking passages as retrieval units [16].

In passage-level methods, documents are usually pre-segmented into small passages. Callan [4] studied how passages can be defined, and how passage signals can be incorporated into document retrieval. In [20], Liu et al. computed a language model for each passage as relevance signals. The final assessment is made by choosing the highest score from all the passages. Their results showed passage-based retrieval can provide more reliable performance than
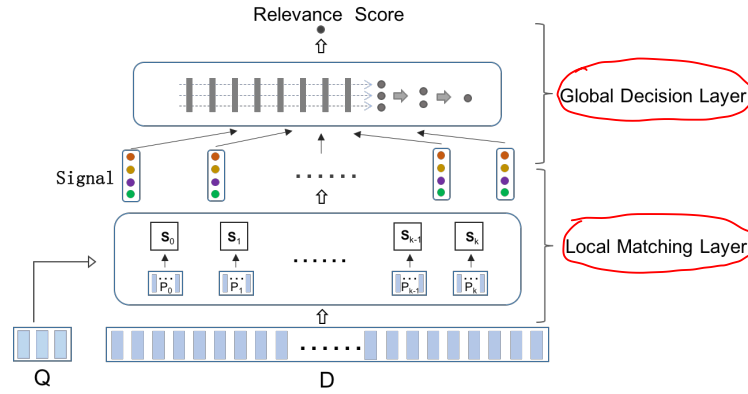
**Figure 1: The Architecture of the Hierarchical Neural Matching Model.**

full document retrieval. In [21], Lv et al. proposed a positional language model in which the relevance score at each position can propagate to nearby positions within certain distance. In other words, each position can be viewed as a "soft passage" by aggregating language model scores within a context window. Based on the signals at each position, they proposed three strategies, namely *best position strategy*, *multi-position strategy* and *multi-σ strategy*, for final relevance assessment.

As we can see, passages are convenient text units for local signal collection to support flexible relevance assessment over documents. However, previous passage-level methods often employed simplified aggregation strategies, thus cannot well capture the diverse relevance patterns for different query-document pairs.

### 2.3 Hybrid Approaches

There also have been a few efforts [2, 35, 37] trying to combine both document-wide and passage-based methods. For example, Callan [4] conducted experiments on four TREC 1 and 2 collections and concluded that it was always better to combine document-wide scores and passage-level scores. A later study by Xi et al. [37] re-examined fixed-size window passages on TREC 4 and 5. Contrary to Callan [4], they did not obtain an improvement by linearly combine passage-level score and document-level score. Wang et al. [35] proposed a discriminative probabilistic model in capturing passage-level signals, and combined the document-level scores and passage-level scores through a heuristic function (i.e., CombMNZ function [19]). However, by using a unified combination strategy, these models cannot fully capture the diverse relevance patterns in different query-document pairs, leading to the mixed performance on different datasets [35, 37].

## 3 HIERARCHICAL NEURAL MATCHING MODEL

In this work, we introduce a HIerarchical Neural maTching (HiNT) model for ad-hoc retrieval to explicitly model the diverse relevance patterns. In an abstract level, the model consists of two stacked components, namely local matching layer and global decision layer. The local matching layer employs deep matching networks to automatically learn the passage-level relevance signals ; The global decision

layer accumulates passage-level signals into different granularities and allows them to compete with each other to generate the final relevance score. The architecture is illustrated in Figure 1. We will describe these components in detail in the follows.

### 3.1 Local Matching Layer

The local matching layer focuses on producing a set of passage-level relevance signals by modeling the relevance matching between a query and each passage of a document. Formally, each document **D** is first represented as a set of passages $\mathbf{D} = [P_1, P_2, ..., P_K]$, where $K$ denotes the number of passages in a document. Then, a set of passage-level relevance signals $\mathbf{E} = [\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_K]$ are produced by applying some relevance matching model $f$ over a query $Q$ and each passage $P_i$.

$$e_i = f(P_i, Q), \qquad i = 1, \ldots, K.$$

There are two major questions concerning this layer, i.e., how to define the passage $P_i$ and how to define the relevance matching model $f$. For passages, there have been three types of definitions: discourse, semantic, and window [4]. Discourse passages are defined based upon textual discourse units (e.g., sentences, paragraphs, and sections). Semantic passages are based upon the subject or content of the text (e.g., TextTiling). Window passages are obtained based upon a number of words. Among these methods, window passage is the most widely adopted due to its simplicity but surprisingly effectiveness as demonstrated by many previous passage-level retrieval models [2, 4, 31, 35].

For the relevance matching model, in general, any model that can address the relevance matching between a query and a passage can be leveraged here. For example, one may employ statistical language model [39], or use manually defined features [28, 36], or even employ some deep models for text matching [14, 34]. However, the quality of the passage-level signals produced by the relevance matching model is a critical foundation for the final relevance assessment. Therefore, we argue that a relevance matching model should be able to encode many well-known IR heuristics to be qualified in this layer. According to previous studies, such heuristics at least include the modeling of exact matching and semantic matching [7, 10], proximity [33], term importance [10] and so on.
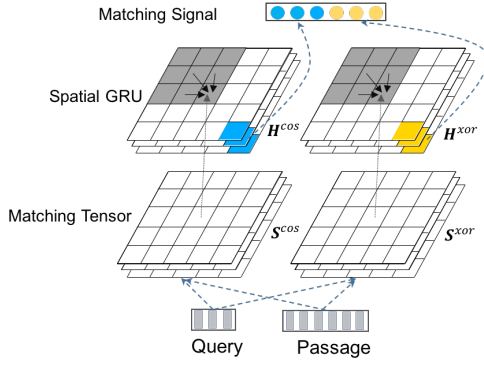
**Figure 2: The Architecture of Relevance Matching Network.**

Based on these above ideas, in this work, we propose to use fixed-size window to define the passage, and employ an existing spatial GRU model [34] for the relevance matching between a query and each passage. This spatial GRU model is good at modeling the matching between two pieces of texts based on primitive word features, and has shown better performances as compared with other deep matching models [34]. We now describe the specific implementation in the follows.

*3.1.1 The Input Layer.* Following the idea in [11], term vectors are employed as basic representations so that rich semantic relations between query and document terms can be captured. Formally, both query and document are represented as a sequence of term vectors denoted by $\mathbf{Q} = [\mathbf{w}_1^{(Q)}, ..., \mathbf{w}_M^{(Q)}]$ and $\mathbf{D} = [\mathbf{w}_1^{(D)}, ..., \mathbf{w}_N^{(D)}]$ where $\mathbf{w}_i^{(Q)}, i = 1, ..., M$ and $\mathbf{w}_j^{(D)}, j = 1, ..., N$ denotes a query term vector and a document term vector, respectively. To obtain the passages, we follow previous approaches [2, 4, 35] to use fixed-size sliding window to segment the document into passages. In this way, the passage is defined as $\mathbf{P} = [\mathbf{w}_1^{(P)}, ..., \mathbf{w}_L^{(P)}]$, where $L$ denotes the window size.

*3.1.2 Deep Relevance Matching Network.* The architecture of the relevance matching network is shown in Figure 2. Firstly, the term-level interaction matrix is constructed based on the term vectors from the query-passage pair. Here we constructed two matching matrices, a semantic matching matrix $\mathbf{M}^{cos}$ and an exact matching matrix (i.e., xor-matrix) $\mathbf{M}^{xor}$, defined as follows:

$$\mathbf{M}_{ij}^{cos} = \frac{\mathbf{w}_i^{(Q)} \mathbf{w}_j^{(P)}}{|\mathbf{w}_i^{(Q)}| \cdot |\mathbf{w}_j^{(P)}|},$$

$$\mathbf{M}_{ij}^{xor} = \begin{cases} 1, & if\ \mathbf{w}_i^{(Q)} = \mathbf{w}_j^{(P)} \\ 0, & otherwise \end{cases}.$$

The key idea of two input matrices is to distinguish the exact matching signals from the semantic matching signals explicitly since the former provides critical information for ad-hoc retrieval as suggested by [7, 10]. Note that in $\mathbf{M}^{cos}$ exact matching and semantic matching signals are mixed together. To further incorporate term importance, we extend each element of $\mathbf{M}_{ij}$ to a three-dimensional

vector $\mathbf{S}_{ij} = [\mathbf{x}_i, \mathbf{y}_j, \mathbf{M}_{ij}]$ by concatenating two corresponding compressed term embeddings as in [27], where $\mathbf{x}_i = \mathbf{w}_i^Q * \mathbf{W}_s$ and $\mathbf{y}_j = \mathbf{w}_j^{(P)} * \mathbf{W}_s$, here, $\mathbf{W}_s$ is the transformation parameter to be learned.

Based on these two query-passage interaction tensors, a spatial GRU (Gated Recurrent Units) is applied to generate the relevance matching evidences. The spatial GRU, also referred to as 2-dimensional Gated-RNN, is a special case of multidimensional RNN [9]. It is a recursive model which scans the input tensor from top left to bottom right:

$$\overrightarrow{\mathbf{H}}_{ij}^{cos} = g(\overrightarrow{\mathbf{H}}_{i-1,j}^{cos}, \overrightarrow{\mathbf{H}}_{i,j-1}^{cos}, \overrightarrow{\mathbf{H}}_{i-1,j-1}^{cos}, \mathbf{S}_{ij}^{cos}),$$

$$\overrightarrow{\mathbf{H}}_{ij}^{xor} = g(\overrightarrow{\mathbf{H}}_{i-1,j}^{xor}, \overrightarrow{\mathbf{H}}_{i,j-1}^{xor}, \overrightarrow{\mathbf{H}}_{i-1,j-1}^{xor}, \mathbf{S}_{ij}^{xor}),$$

where $g$ denotes the spatial GRU unit as described in [34], $\overrightarrow{\mathbf{H}}_{ij}^{cos}$ and $\overrightarrow{\mathbf{H}}_{ij}^{xor}$ denotes the hidden state of the spatial GRU over $\mathbf{S}^{cos}$ and $\mathbf{S}^{xor}$, respectively. We can take the last hidden representation $\mathbf{H}_{M,L}$ as the matching output. The local relevance evidence $\overrightarrow{\mathbf{e}}$ is then generated by concatenating the two matching outputs:

$$\overrightarrow{\mathbf{e}} = [\overrightarrow{\mathbf{H}}_{M,L}^{cos}, \overrightarrow{\mathbf{H}}_{M,L}^{xor}].$$

Furthermore, in order to enrich the relevance signals, we also applied the spatial GRU in the reverse direction, i.e., from bottom right to top left. The final passage-level signal is defined as the concatenation of the two-direction matching signals:

$$\mathbf{e} = [\overrightarrow{\mathbf{e}}, \overleftarrow{\mathbf{e}}].$$

## 3.2 Global Decision Layer

Based on passage-level signals generated in the previous step, the global decision layer attempts to accumulate these signals into different granularities and allow them to compete with each other for final relevance assessment. As we have discussed before, the relevance patterns of a query-document pair can be rather flexible and diverse, allowing a document to be relevant to a query partially or as a whole. Accordingly, the global decision layer is expected to be able to accommodate various decision strategies, rather than using some restrictive combination rules [2, 35].

In this work, we propose to employ a hybrid neural network architecture which has sufficient expressive power to support flexible relevance patterns. Before we describe the hybrid model, we first introduce two basic models under some simplified relevance assumptions.

1. **Independent Decision (ID) Model** assumes the independence among passage-level signals, and selects top-k signals directly for final relevance assessment. This model is under the assumption that a document is relevant if any piece of it can provide sufficient relevance information. Specifically, as shown in Figure 3(a), a dimension-wise k-max pooling layer is first applied over the passage-level signals to select top-k signals, and the selected signals are then concatenated and projected into a multi-layer perceptron to get the final decision score.

2. **Accumulative decision (AD) Model** accumulates the passage-level signals in a sequential manner, and selects top-k accumulated signals for relevance assessment. Here,
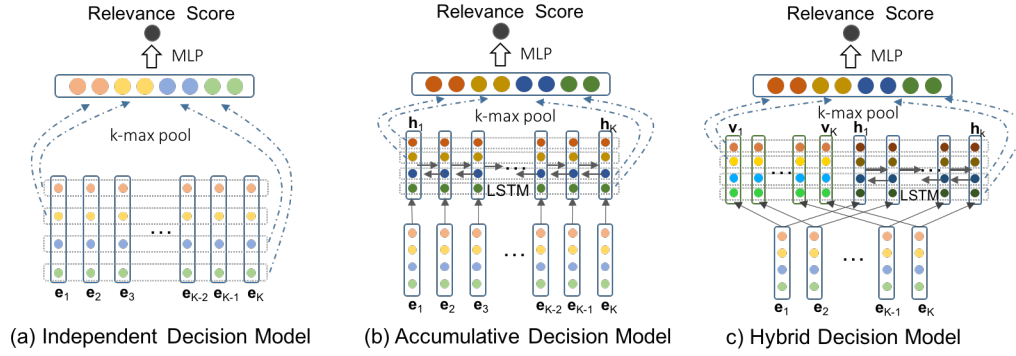
**Figure 3: Different decision models based on the collected passage signals.**

we adopt long-short term memory network (LSTM) [12], a powerful model for variable-length sequential data, to accumulate the relevance signals from each passage. Specifically, as shown in Figure 3(b), we first feed the passage-level signals into LSTM sequentially to generate the accumulated relevance signals at different positions.Based on the accumulated relevance signals, we then apply a dimension-wise k-max pooling layer to select top-k signals, and feed the selected signals into a multi-layer perceptron for final relevance assessment. Here, we also applied the LSTM in the reverse direction to accumulate the relevance signals, as user's reading can be in any direction of the document[29]. Note here if we directly use the last/first hidden state in LSTM as signals for relevance assessment, it would reduce to a document-wide method. By using k-max pooling over all the positions, we actually assume the relevance could be based on a text span flexible in scale, ranging from multiple passages to the whole document.

Based on the two basic models, now we introduce the **Hybrid Decision (HD) Model** as a specific implementation of the global decision layer. The HD model is a mixture of the previous two models, and picks top-k signals from passage-level or accumulated signals for final relevance assessment. Obviously, this is the most flexible relevance model, which allows a document to be assessed as a whole or partially adaptively. Specifically, as depicted in figure 3(c), we allow the relevance signals from different passages to compete with accumulated signals. Note here in order to make a fair competition, for the passage-level signals, we conduct an additional non-linear transformation to ensure a similar scale to the accumulated relevance signals.

$$\mathbf{v}_t = \tanh(\mathbf{W}_v \mathbf{e}_t + \mathbf{b}_v),$$

where $\mathbf{v}_t$ denotes the $t$-th transformed passage signals, $\mathbf{W}_v$ and $\mathbf{b}_v$ are parameters to be learned. We then apply a dimension-wise k-max pooling layer to select top-k signals, and feed the selected signals into a multi-layer perceptron for final assessment.

## 3.3 Model Training

Since the ad-hoc retrieval task is fundamentally a ranking problem, we utilize the pairwise ranking loss such as hinge loss to train our

**Table 1: Statistics of the datasets used in this study.**

|  | #queries | #docs | #q_rel | #rel_per_q |
|---|---|---|---|---|
| MQ2007 | 1692 | 65,323 | 1455 | 10.3 |
| MQ2008 | 784 | 14,384 | 564 | 3.7 |

model. Specifically, given a triple $(q, d^+, d^-)$, where $d^+$ is ranked higher than $d^-$ with respect to a query $q$, the loss function is defined as:

$$\mathcal{L}(q, d^+, d^-; \theta) = \max(0, 1 - s(q, d^+) + s(q, d^-)),$$

where $s(q, d)$ denotes the relevance score for $(q, d)$, and $\theta$ includes the parameters in both local matching layer and global decision layer. The optimization is relatively straightforward with standard backpropagation. We apply stochastic gradient decent method Adam [17] with mini-batches(100 in size), which can be easily parallelized on a single machine with multi-cores.

## 4 EXPERIMENT

In this section, we conduct experiments to demonstrate the effectiveness of our proposed model on benchmark collections.

## 4.1 Experimental Settings

We first introduce our experimental settings, including datasets, baseline methods/implementations, and evaluation methodology.

*4.1.1 Data Sets.* To evaluate the performance of our model, we conducted experiments on two LETOR benchmark datasets [28]: Million Query Track 2007 (MQ2007) and Million Query Track 2008 (MQ2008). We choose these two datasets according to three criteria: 1) there is a large number of queries, 2) the original document content is available, and 3) the dataset is public. The first two criterias are important for learning deep neural models for ad-hoc retrieval, and the last one is critical for reproducibility. Both datasets use the GOV2 collection which includes 25 million documents in 426 gigabytes. The details of the two datasets are given in Table 1. As we can see, there are 1692 queries on MQ2007 and 784 queries on MQ2008. The number of queries with at least one relevant document is 1455 and 564, respectively. The average number of relevant document per query is about 10.3 and 3.7 on MQ2007 and MQ2008, respectively.

For pre-processing, all the words in documents and queries are white-space tokenized, lower-cased, and stemmed using the Krovetz stemmer [18]. Stopword removal is performed on query and document words using the INQUERY stop list [5]. Words occurred less than 5 times in the collection are removed from all the document. We further segmented documents into passages for all the models using passage-level information. We utilized fixed-size sliding window without overlap to generate passages. We have also studied the performance of different window size in Section 4.5.

*4.1.2 Baselines Methods.* We adopt three types of baselines for comparison, including traditional retrieve models, learning to rank models and deep matching models.

For traditional retrieval models, we consider both document-wide methods, passage-level methods, and hybrid methods:

**BM25**: The BM25 model [30] is a classical and highly effective document-wide retrieval model.

**MSP**: The Max-Scoring Passage model [20] utilizes language model for each passage and rank the document according to the score of their best passage.

**PLM**: The passage language model [2] integrates passage-level and document-wide language model scores according to the document homogeneity for ad-hoc retrieval.

**PPM**: The probabilistic passage model [35] is a discriminative probabilistic model in capturing passage-level signals, and combines document retrieval scores with passage retrieval scores through a linear interpolate function.

Learning to rank models include

**AdaRank**: AdaRank [38] is a representative pairwise model which aims to directly optimize the performance measure based on boosting approach. Here we utilize NDCG as the performance measure function.

**LambdaMart**: LambdaMart [3] is a representative listwise model that uses gradient boosting to produce an ensemble of retrieval models. It is the state-of-the-art learning to rank algorithm.

Here, AdaRank and LambdaMart were implemented using RankLib[2], which is a widely adopted learning to rank tool. All the learning to rank models leveraged the 46 human designed features from LETOR. Furthermore, since our model utilized passage-level information, we introduced 9 passage-based features for fair comparison. Specifically, we calculated tf-idf, BM25 and language model scores for each query-passage pair, and picked the maximum, minimum and average scores across passages as the new features for a document. We applied the full set of features (original+passage features) on both two learning to rank models for additional comparison, denoted by **AdaRank(+P)** and **LambdaMart(+P)**, respectively.

Deep matching models include

**DSSM**: DSSM [14] is a neural matching model proposed for Web search. It consists of a word hashing layer, two non-linear hidden layers, and an output layer.

**DRMM**: DRMM [10] is a neural relevance model designed for ad-hoc retrieval. It consists of a matching histogram mapping, a feed forward matching network and a term gating network.

**Duet**: Duet [24] is a joint model which learns local lexical matching and global semantic matching together.

**DeepRank**: DeepRank [27] is a state-of-the-art deep matching model which models relevance by simulating the human judgement process.

Here, for DSSM, we directly utilize the trained model[3] released by their authors since training these complex models on small benchmark datasets could lead to severe over-fitting problem. For DeepRank[4], we use the code released by their authors. For Duet model, we train it by ourselves since there is no trained model released. To avoid overfitting, we reduce the parameters of the convolutional network and fully-connected network to adapt the model to the limited size of LETOR datasets. Specifically, we set the filter size as 10 in both local model and global model, and the hidden size as 20 in the fully-connected layer. Other parameters are the same as the original paper.

We refer to our proposed model as **HiNT**[5]. For network configurations (e.g., numbers of layers and hidden nodes), we tuned the hyper-parameters via the validation set. Specifically, in the local matching layer, the dimension of the spatial GRU is set to 2 which is tuned in [1, 2, 3, 4]. In the global decision layer, the dimension of LSTM is set to 6 which is tuned in [4, 5, 6, 7, 8, 9, 10], the k-max pooling size is set to 10 which is tuned in [1, 5, 10, 15, 20], and the multi-layer perceptron is a 2-layers feed forward network without hidden layers. All the trainable parameters are initialized randomly by uniform distribution within $[-0.1, 0.1]$. Overall, the number of trainable parameters is about 930 in our HiNG model. Note that the MQ2008 dataset has much smaller query and document size, we find it is not sufficient to train deep models purely based on this dataset. Therefore, for all the deep models, we chose to use the trained model on MQ2007 as the initialization and fine tuned the model on MQ2008.

For all deep models based on term vector inputs, we used 50-dimension term vectors. The term vectors were trained on wikipedia corpus[6] using the CBOW model [22] with the default parameters[7]. Specifically, we used 10 as the context window size, 10 negative samples and a subsampling of frequent words with sampling threshold of $10^{-4}$. Out-of-vocabulary words are randomly initialized by sampling values uniformly from $(-0.02, 0.02)$.

*4.1.3 Evaluation Methodology.* We follow the data partition on this dataset in Letor4.0 [28], and 5 fold cross-validation is conducted to minimize over-fitting as in [10]. Specifically, the parameters for each model are tuned on 4-of-5 folds. The last fold in each case is used for evaluation. This process is repeated 5 times, once for each fold. The results reported are the average over the 5 folds.

As for evaluation measure, precision (P), mean average precision (MAP) and normalized discounted cumulative gain (NDCG) at position 1, 5, and 10 were used in our experiments. We performed significant tests using the paired t-test. Differences are considered statistically significant when the $p-$value is lower than 0.05.

---

[3]https://www.microsoft.com/en-us/research/project/dssm/
[4]https://github.com/pl8787/textnet-release
[5]https://github.com/faneshion/HiNT
[6]http://en.wikipedia.org/wiki/Wikipedia_database
[7]https://code.google.com/archive/p/word2vec/

**Table 2: Analysis on local matching layer on MQ2007. Significant improvement or degradation with respect to our implementation ($S^{xor}$+$S^{cos}$+spatial GRU) is indicated (+/-) (p-value $\leq 0.05$)**

| Local Matching Layer | IR Heuristics | | | | | Performance | | |
|---|---|---|---|---|---|---|---|---|
| | Exact matching | Semantic matching | Exact/Semantic Distinguished | Proximity | Term Importance | P@10 | NDCG@10 | MAP |
| $\mathbf{M}^{xor}$+MLP | √ | | | | | $0.384^-$ | $0.435^-$ | $0.461^-$ |
| $\mathbf{M}^{cos}$+MLP | √ | √ | | | | $0.329^-$ | $0.344^-$ | $0.386^-$ |
| $\mathbf{M}^{hist}$+MLP | √ | √ | √ | | | $0.393^-$ | $0.447^-$ | $0.469^-$ |
| $\mathbf{M}^{xor}$+spatial GRU | √ | | | √ | | $0.387^-$ | $0.444^-$ | $0.465^-$ |
| $\mathbf{M}^{cos}$+spatial GRU | √ | √ | | √ | | $0.396^-$ | $0.449^-$ | $0.470^-$ |
| $\mathbf{M}^{xor}$+$\mathbf{M}^{cos}$+spatial GRU | √ | √ | √ | √ | | $0.405^-$ | $0.470^-$ | $0.484^-$ |
| $S^{xor}$+$S^{cos}$+spatial GRU | √ | √ | √ | √ | √ | 0.418 | 0.490 | 0.502 |

## 4.2 Analysis on the HiNT Model

In this section we conducted experiments to compare different implementations of the two components in the HiNT model. Through these experiments, we try to gain a better understanding of the model.

### 4.2.1 Analysis on Local Matching Layer.
As mentioned in Section 3.1, the local matching layer should be able to encode many well-known IR heuristics in order to well capture the relevance matching between a query and a passage. The heuristics at least include the modeling of exact matching and semantic matching signals, the differentiation between them, the modeling of proximity, the term importance, and so on. Here we conduct experiments to test a variety of implementations of the local matching layer which encode different IR heuristics by fixing the rest parts of the model.

The implementations include: (1) We apply a multi-layer perceptron over the exact matching matrix $\mathbf{M}^{xor}$ to produce the passage-level signals. In this way, only exact matching signals are encoded into the passage-level signal; (2) We apply a multi-layer perceptron over the semantic matching matrix $\mathbf{M}^{cos}$ to produce the passage-level signals. In this way, both exact and semantic matching signals are encoded but mixed together; (3) We follow the idea in [10] to turn the semantic matching matrix $\mathbf{M}^{cos}$ into matching histograms $\mathbf{M}^{hist}$, and use a multi-layer perceptron to produce the passage-level signals. In this way, both exact matching and semantic matching signals are encoded and these two types of signals are differentiated by using the histogram; (4) We apply a spatial GRU over the exact matching matrix $\mathbf{M}^{xor}$ to produce the passage-level signals. In this way, only exact matching signals and proximity are encoded into the signals; (5) We apply a spatial GRU over the semantic matching matrix $\mathbf{M}^{cos}$ to produce the passage-level signals. In this way, exact matching and semantic matching signals are mixed and encoded together with proximity information. (6) We use a spatial GRU over both exact matching and semantic matching signals. Here, the exact matching and semantic matching signals can be clearly differentiated. Finally, we use our proposed implementation, i.e., a spatial GRU over the $S^{xor}$ and $S^{cos}$ tensors, which can encode exact matching signals, semantic matching signals, proximity and term importance. The different implementations of the local matching layer as well as their performance results are shown in Table 2.

From the results we observe that, when modeling exact matching signals alone, $\mathbf{M}^{xor} + MLP$ can already obtain reasonably good retrieval performance. It indicates that exact matching signals are
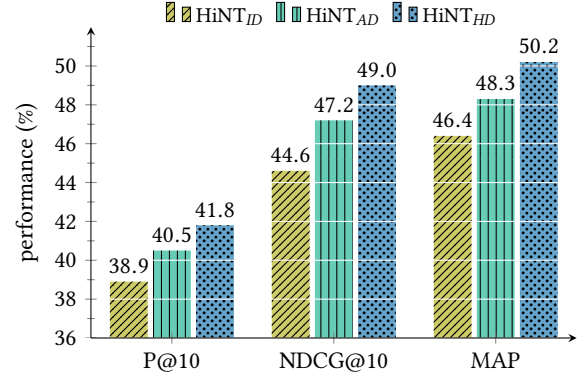


**Figure 4: Performance comparison of HiNT over different decision models on MQ2007.**

critical in ad-hoc retrieval [10]. Meanwhile, the performance drops significantly when semantic matching signals are mixed with exact matching signals ($\mathbf{M}^{cos} + MLP$), but increases if these two types of signals are clearly differentiated ($\mathbf{M}^{hist} + MLP$). These results demonstrate that semantic matching signals are also useful for retrieval, but should better be distinguished from exact matching signals if the deep model itself (e.g., MLP) cannot differentiate them. If the deep model can somehow implicitly distinguish these two types of signals, e.g., spatial GRU using input gates, we can observe better performance on the semantic matching matrix ($\mathbf{M}^{cos} + spatialGRU$) than that on the exact matching matrix ($\mathbf{M}^{xor} + spatialGRU$). However, we can further observe performance increase if we explicitly distinguish exact matching and semantic matching signals ($\mathbf{M}^{xor} + \mathbf{M}^{cos} + spatialGRU$). Besides, the local matching layers using spatial GRU can in general obtain better results, indicating that proximity is very helpful for retrieval. Finally, by further considering term importance, our proposed implementation ($S^{xor} + S^{cos} + spatialGRU$) can outperform all the variants significantly. All the results demonstrate the importance of encoding a variety of IR heuristics in the local matching layer for a successful relevance judgement model.

### 4.2.2 Analysis on Global Decision Layer.
We further study the effect of different implementations of the global decision layer. Here we compare the proposed hybrid decision model (i.e., HiNT$_{HD}$) with the two basic decision models introduced in Section 3.2 (i.e.,

**Table 3: Comparison of different retrieval models over the MQ2007 and MQ2008 datasets. Significant improvement or degradation with respect to HiNT is indicated (+/-) (p-value $\leq 0.05$).**

| | | | | MQ2007 | | | |
|---|---|---|---|---|---|---|---|
| Model Name | P@1 | P@5 | P@10 | NDCG@1 | NDCG@5 | NDCG@10 | MAP |
| BM25 | 0.427⁻ | 0.388⁻ | 0.366⁻ | 0.358⁻ | 0.384⁻ | 0.414⁻ | 0.450⁻ |
| MSP | 0.361⁻ | 0.358⁻ | 0.350⁻ | 0.302⁻ | 0.341⁻ | 0.378⁻ | 0.422⁻ |
| PLM | 0.416⁻ | 0.389⁻ | 0.371⁻ | 0.348⁻ | 0.377⁻ | 0.413⁻ | 0.449⁻ |
| PPM | 0.431⁻ | 0.393⁻ | 0.370⁻ | 0.361⁻ | 0.392⁻ | 0.424⁻ | 0.453⁻ |
| AdaRank | 0.449⁻ | 0.403⁻ | 0.372⁻ | 0.394⁻ | 0.410⁻ | 0.436⁻ | 0.460⁻ |
| LambdaMart | 0.481⁻ | 0.418⁻ | 0.384⁻ | 0.412⁻ | 0.421⁻ | 0.446⁻ | 0.468⁻ |
| AdaRank(+P) | 0.457⁻ | 0.408⁻ | 0.380⁻ | 0.393⁻ | 0.408⁻ | 0.438⁻ | 0.467⁻ |
| LambdaMart(+P) | 0.484⁻ | 0.427⁻ | 0.391⁻ | 0.413⁻ | 0.427⁻ | 0.454⁻ | 0.473⁻ |
| DSSM | 0.345⁻ | 0.359⁻ | 0.352⁻ | 0.290⁻ | 0.335⁻ | 0.371⁻ | 0.409⁻ |
| DRMM | 0.450⁻ | 0.417⁻ | 0.388⁻ | 0.380⁻ | 0.408⁻ | 0.440⁻ | 0.467⁻ |
| Duet | 0.473⁻ | 0.428⁻ | 0.398⁻ | 0.409⁻ | 0.431⁻ | 0.453⁻ | 0.474⁻ |
| DeepRank | 0.508 | 0.452⁻ | 0.412⁻ | 0.441 | 0.457⁻ | 0.482⁻ | 0.497 |
| **HiNT** | **0.515** | **0.461** | **0.418** | **0.447** | **0.463** | **0.490** | **0.502** |
| | | | | MQ2008 | | | |
| Model Name | P@1 | P@5 | P@10 | NDCG@1 | NDCG@5 | NDCG@10 | MAP |
| BM25 | 0.408⁻ | 0.337⁻ | 0.245 | 0.344⁻ | 0.461⁻ | 0.220⁻ | 0.465⁻ |
| MSP | 0.332⁻ | 0.314⁻ | 0.236⁻ | 0.283⁻ | 0.415⁻ | 0.193⁻ | 0.426⁻ |
| PLM | 0.396⁻ | 0.326⁻ | 0.240⁻ | 0.327⁻ | 0.438⁻ | 0.208⁻ | 0.452⁻ |
| PPM | 0.412⁻ | 0.338⁻ | 0.241⁻ | 0.350⁻ | 0.464⁻ | 0.220⁻ | 0.468⁻ |
| AdaRank | 0.434⁻ | 0.342 | 0.243 | 0.368⁻ | 0.468 | 0.221 | 0.476 |
| LambdaMart | 0.449⁻ | 0.346 | 0.249 | 0.376⁻ | 0.471 | 0.230 | 0.478 |
| AdaRank(+P) | 0.428⁻ | 0.345 | 0.247 | 0.368⁻ | 0.475 | 0.225 | 0.478 |
| LambdaMart(+P) | 0.441⁻ | 0.348 | 0.249 | 0.372⁻ | 0.479 | 0.232 | 0.480 |
| DSSM | 0.341⁻ | 0.284⁻ | 0.221⁻ | 0.286⁻ | 0.378⁻ | 0.178⁻ | 0.391⁻ |
| DRMM | 0.450⁻ | 0.337⁻ | 0.242⁻ | 0.381⁻ | 0.466⁻ | 0.219⁻ | 0.473⁻ |
| Duet | 0.452⁻ | 0.341⁻ | 0.240⁻ | 0.385⁻ | 0.471⁻ | 0.216 | 0.476⁻ |
| DeepRank | 0.482⁻ | 0.359⁻ | 0.252 | 0.406⁻ | 0.496 | 0.240 | 0.498⁻ |
| **HiNT** | **0.491** | **0.367** | **0.255** | **0.415** | **0.501** | **0.244** | **0.505** |

$HiNT_{ID}$ and $HiNT_{AD}$) by fixing the rest parts of the model. The comparison results are shown in Figure 4. As we can see, the simplest relevance model $HiNT_{ID}$ performs worst. It seems that selecting passage-level signals independently might be too simple to capture diverse relevance patterns. Meanwhile, $HiNT_{AD}$ performs better than $HiNT_{ID}$, indicating that it is more beneficial to make relevance assessment based on accumulated signals from a variety of text spans. Finally, $HiNT_{HD}$ achieves the best performance in terms of all the evaluation measures. This further indicates that there might be very diverse relevance patterns across different query-document pairs. By allowing competition between passage-level and accumulated signals, the expressive power of the HD model is the largest, leading to the best performance among the three variants.

### 4.3 Comparison of Retrieval Models

In this section, we compare our proposed HiNT against existing retrieval models over the two benchmark datasets. Note here we refer HiNT to the model using hybrid decision model based on both exact matching and semantic matching tensors. The main results of our experiments are summarized in Table 3.

Firstly, for the traditional models, we can see that BM25 is a strong baseline which performs better than MSP. The relative poor performance of MSP indicates that it is deficient in capturing the diverse relevance patterns by only using the passage-level signals.

By integrating document-wide with passage-level signals, the performance of PLM and PPM is mixed compared with BM25, demonstrating the deficiency of the simple combination strategy, which is consistent with previous findings [35]. Secondly, all the learning to rank models perform significantly better than the traditional retrieval models. This is not surprising since learning to rank models can make use of rich features, where BM25 scores and LM scores are typical features among them. Among the two learning to rank models, LambdaMart performs better. Moreover, we can see that adding passage-level features might improve the performance of learning to rank models, but not consistent on different datasets. For example, the performance of AdaRank and LambdaMart in terms of P@1 on MQ2008 drops when adding passage features. Thirdly, as for the deep matching models, we can see that DSSM obtain relatively poor performances on both datasets, even cannot compete with the traditional retrieval models. This is consistent with previous findings [10], showing that one may not work well on ad-hoc retrieval by only leveraging the cosine similarity between high-level abstract representations of short query and long document. As for DRMM and Duet, they have achieved relative better performance compared with DSSM. This may due to the fact that they are specifically designed for the relevance matching in ad-hoc retrieval, and they have incorporated important IR heuristics into their model. However, they can only reach comparable performance as learning to rank models by only using document-wide
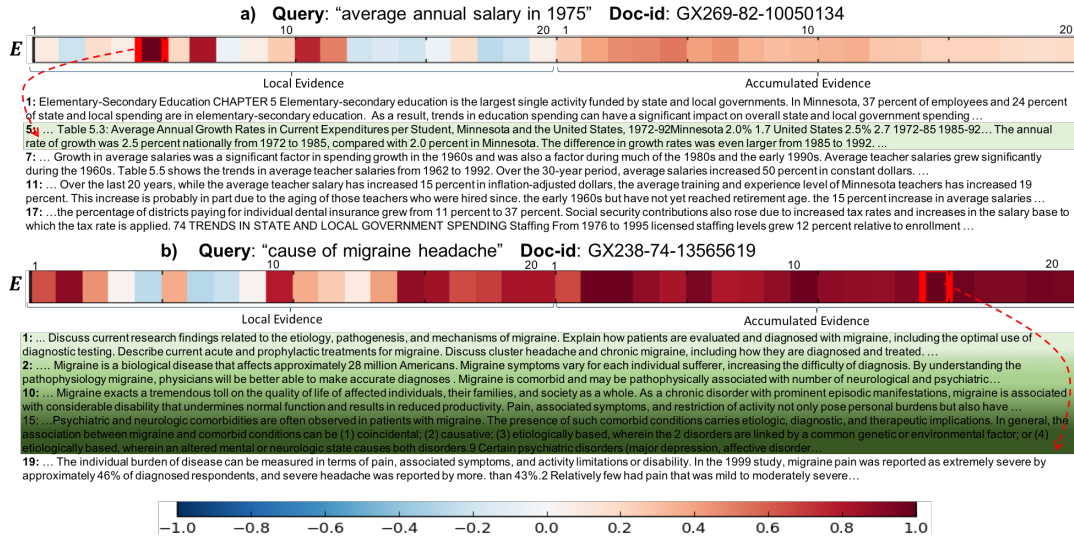
**a) Query**: "average annual salary in 1975"  **Doc-id**: GX269-82-10050134

**Local Evidence**

**Accumulated Evidence**

**1:** Elementary-Secondary Education CHAPTER 5 Elementary-secondary education is the largest single activity funded by state and local governments. In Minnesota, 37 percent of employees and 24 percent of state and local spending are in elementary-secondary education. As a result, trends in education spending can have a significant impact on overall state and local government spending …

**5:** … Table 5.3: Average Annual Growth Rates in Current Expenditures per Student, Minnesota and the United States, 1972-92Minnesota 2.0% 1.7 United States 2.5% 2.7 1972-85 1985-92… The annual rate of growth was 2.5 percent nationally from 1972 to 1985, compared with 2.0 percent in Minnesota. The difference in growth rates was even larger from 1985 to 1992. …

**7:** … Growth in average salaries was a significant factor in spending growth in the 1960s and was also a factor during much of the 1980s and the early 1990s. Average teacher salaries grew significantly during the 1960s. Table 5.5 shows the trends in average teacher salaries from 1962 to 1992. Over the 30-year period, average salaries increased 50 percent in constant dollars. …

**11:** … Over the last 20 years, while the average teacher salary has increased 15 percent in inflation-adjusted dollars, the average training and experience level of Minnesota teachers has increased 19 percent. This increase is probably in part due to the aging of those teachers who were hired since. the early 1960s but have not yet reached retirement age. the 15 percent increase in average salaries …

**17:** …the percentage of districts paying for individual dental insurance grew from 11 percent to 37 percent. Social security contributions also rose due to increased tax rates and increases in the salary base to which the tax rate is applied. 74 TRENDS IN STATE AND LOCAL GOVERNMENT SPENDING Staffing From 1976 to 1995 licensed staffing levels grew 12 percent relative to enrollment …

**b) Query**: "cause of migraine headache"  **Doc-id**: GX238-74-13565619

**Local Evidence**

**Accumulated Evidence**

**1:** … Discuss current research findings related to the etiology, pathogenesis, and mechanisms of migraine. Explain how patients are evaluated and diagnosed with migraine, including the optimal use of diagnostic testing. Describe current acute and prophylactic treatments for migraine. Discuss cluster headache and chronic migraine, including how they are diagnosed and treated. …

**2:** …. Migraine is a biological disease that affects approximately 28 million Americans. Migraine symptoms vary for each individual sufferer, increasing the difficulty of diagnosis. By understanding the pathophysiology migraine, physicians will be better able to make accurate diagnoses . Migraine is comorbid and may be pathophysically associated with number of neurological and psychiatric…

**10:** … Migraine exacts a tremendous toll on the quality of life of affected individuals, their families, and society as a whole. As a chronic disorder with prominent episodic manifestations, migraine is associated with considerable disability that undermines normal function and results in reduced productivity. Pain, associated symptoms, and restriction of activity not only pose personal burdens but also have …

**15:** …Psychiatric and neurologic comorbidities are often observed in patients with migraine. The presence of such comorbid conditions carries etiologic, diagnostic, and therapeutic implications. In general, the association between migraine and comorbid conditions can be (1) coincidental; (2) causative; (3) etiologically based, wherein the 2 disorders are linked by a common genetic or environmental factor; or (4) etiologically based, wherein an altered mental or neurologic state causes both disorders.9 Certain psychiatric disorders (major depression, affective disorder…

**19:** … The individual burden of disease can be measured in terms of pain, associated symptoms, and activity limitations or disability. In the 1999 study, migraine pain was reported as extremely severe by approximately 46% of diagnosed respondents, and severe headache was reported by more. than 43%.2 Relatively few had pain that was mild to moderately severe…

**Figure 5: Examples of different decision strategies over the query-document pair.**

matching signals. The recently introduced DeepRank achieves a relative better performance by simulating the human judgement process. However, DeepRank aggregates relevance signals from query-centric contexts to form document-wide relevance score for each query term, it cannot well capture the diverse relevance patterns between query-document pairs.

Finally, we observe that HiNT can outperform all the existing models in terms of all the evaluation measures on both datasets by allowing the competition between the passage-level signals and document-wide signals explicitly. It is worth noting that in the learning to rank methods, there are many human designed features including those document-wide and passage-level matching signals, as well as those about the quality of the document (e.g., PageRank). While in our HiNT, all the assessment are purely learned from the primitive word features of queries and documents. Therefore, the superior performance of HiNT suggests the importance and effectiveness of modeling the diverse relevance patterns for ad-hoc retrieval.

## 4.4 Impact of Passage Size

Since we leverage the fixed-size sliding windows to segment a document into passages, we would like to study the effect of the passage size on the ranking performance. Here we report the performance results on MQ2007 with the passage size set as 10, 50, 100, 200, 300 and 500 words. As shown in Figure 6, the performance first increases and then drops with the increase of the passage size. The possible reason might be that too small passage size may hurt the quality of passage signals (i.e., relevance matching between the query and the passage) due to the information sparsity, while too large passage size would produce limited number of coarse passage-level signals which restrict the ability of the global decision layer. Our results shows that the best performance can be achieved when the passage size is set to 100 words on MQ2007.
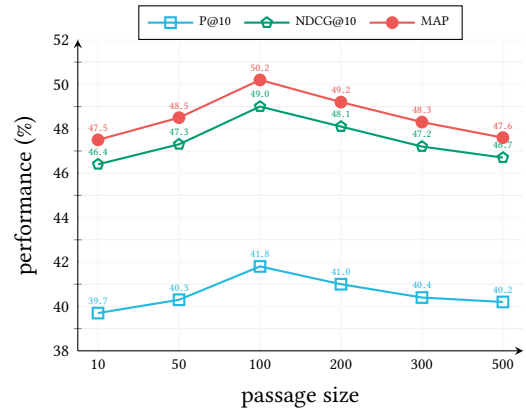
**Figure 6: Performance comparison of HiNT over different passage sizes on MQ2007.**

## 4.5 Case Study

To better understand what can be learned by HiNT, here we conduct some case studies. For better visualization and analysis, we simplified our model by replacing k-max pooling with max pooling so that only the most significant signal is used in decision. Based on the learned model, we pick up a query and a relevant document, and plot all the signals **E** used in the hybrid decision model along with the corresponding document content. Here each small bar in **E** denotes a passage or accumulated signal at that position, with the color corresponding to the signal strength. We highlight the final selected signal with a red box and the corresponding passages with green background color.

As shown in Figure 5, we can find two significantly different decision strategies between a query and a document. In the first case, the document is relevant to a query because of a strong passage signal. By checking the query and the document, we find that

the query is "average annual salary in 1975" which conveys very specific information need, and the passage at the 5-th position (i.e., the strongest signal) contains a table whose content can well address this information need. In the second case, the document is relevant to a query because of a strong accumulated signal. Again by checking the query and the document, we find that the query is about the "cause of migraine headache" which is informational, and the document is mostly relevant to this query with many passages addressing this problem (i.e., from the beginning to the 15-th passage).

The two cases show that there are indeed quite diverse relevance patterns in real-world retrieval scenario, and our HiNT model can capture these diverse relevance patterns successfully.

## 5 CONCLUSIONS

In this paper, we have introduced a hierarchical neural matching model to capture the diverse relevance patterns in ad-hoc retrieval. The model consists of two components, namely local matching layer and global decision layer. We employed deep neural network in both layers to support high-quality relevance signal generation and flexible relevance assessment strategies, respectively. Experimental results on two benchmark datasets demonstrate that our model can outperform all the baseline models in terms to all the evaluation metrics, especially state-of-the-art learning to rank methods that use manually designed features.

For future work, it would be interesting to try other implementation of the two components in HiNT, e.g., to employ some attention-based neural network for the global decision layer. We would also like to expand our model to accommodate features beyond relevance matching, e.g. PageRank, to help improve the retrieval performance.

## 6 ACKNOWLEDGMENTS

## REFERENCES

[1] Gianni Amati and Cornelis Joost Van Rijsbergen. 2002. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems (TOIS)* 20, 4 (2002), 357–389.
[2] Michael Bendersky and Oren Kurland. 2008. Utilizing passage-based language models for document retrieval. In *European Conference on Information Retrieval*. Springer, 162–174.
[3] Christopher JC Burges. 2010. From ranknet to lambdarank to lambdamart: An overview. *Learning* 11 (2010), 23–581.
[4] James P Callan. 1994. Passage-level evidence in document retrieval. In *SIGIR*. Springer-Verlag New York, Inc., 302–310.
[5] James P Callan, W Bruce Croft, and John Broglio. 1995. TREC and TIPSTER experiments with INQUERY. *Information Processing & Management* 31, 3 (1995), 327–343.
[6] Charles LA Clarke, Falk Scholer, and Ian Soboroff. 2005. The TREC 2005 Terabyte Track.. In *TREC*.
[7] Hui Fang and ChengXiang Zhai. 2006. Semantic term matching in axiomatic approaches to information retrieval. In *SIGIR*. ACM, 115–122.
[8] Yoav Freund, Raj Iyer, Robert E Schapire, and Yoram Singer. 2003. An efficient boosting algorithm for combining preferences. *Journal of machine learning research* 4, Nov (2003), 933–969.
[9] Alex Graves and Jürgen Schmidhuber. 2009. Offline handwriting recognition with multidimensional recurrent neural networks. In *Advances in neural information processing systems*. 545–552.
[10] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. 2016. A deep relevance matching model for ad-hoc retrieval. In *CIKM*. ACM, 55–64.
[11] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. 2016. Semantic matching by non-linear word transportation for information retrieval. In *CIKM*. ACM, 701–710.
[12] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.
[13] Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2014. Convolutional neural network architectures for matching natural language sentences. In *NIPS*. 2042–2050.
[14] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *CIKM*. ACM, 2333–2338.
[15] Thorsten Joachims. 2006. Training linear SVMs in linear time. In *SIGKDD*. ACM, 217–226.
[16] Marcin Kaszkiel and Justin Zobel. 1997. Passage retrieval revisited. In *SIGIR*, Vol. 31. ACM, 178–185.
[17] Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
[18] Robert Krovetz. 1993. Viewing morphology as an inference process. In *SIGIR*. ACM, 191–202.
[19] Joon Ho Lee. 1997. Analyses of multiple evidence combination. In *ACM SIGIR Forum*, Vol. 31. ACM, 267–276.
[20] Xiaoyong Liu and W. Bruce Croft. 2002. Passage retrieval based on language models. In *CIKM*. 375–382.
[21] Yuanhua Lv and Cheng Xiang Zhai. 2009. Positional language models for information retrieval. In *SIGIR*. 299–306.
[22] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.
[23] Bhaskar Mitra and Nick Craswell. 2017. Neural Models for Information Retrieval. *arXiv preprint arXiv:1705.01509* (2017).
[24] Bhaskar Mitra, Fernando Diaz, and Nick Craswell. 2017. Learning to Match using Local and Distributed Representations of Text for Web Search. In *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 1291–1299.
[25] Seung-Hoon Na. 2015. Two-stage document length normalization for information retrieval. *ACM Transactions on Information Systems (TOIS)* 33, 2 (2015), 8.
[26] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, and Xueqi Cheng. 2016. A study of matchpyramid models on ad-hoc retrieval. *arXiv preprint arXiv:1606.04648* (2016).
[27] 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 *CIKM*. ACM, 257–266.
[28] Tao Qin, Tie-Yan Liu, Jun Xu, and Hang Li. 2010. LETOR: A benchmark collection for research on learning to rank for information retrieval. *Information Retrieval* 13, 4 (2010), 346–374.
[29] Keith Rayner. 1998. Eye movements in reading and information processing: 20 years of research. *Psychological bulletin* 124, 3 (1998), 372.
[30] Stephen E Robertson and Steve Walker. 1994. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *SIGIR*. Springer-Verlag New York, Inc., 232–241.
[31] Gerard Salton, James Allan, and Chris Buckley. 1993. Approaches to passage retrieval in full text information systems. In *SIGIR*. ACM, 49–58.
[32] Mark Sanderson. 2010. *Test collection based evaluation of information retrieval systems*. Now Publishers Inc.
[33] Tao Tao and ChengXiang Zhai. 2007. An exploration of proximity measures in information retrieval. In *SIGIR*. ACM, 295–302.
[34] Shengxian Wan, Yanyan Lan, Jun Xu, Jiafeng Guo, Liang Pang, and Xueqi Cheng. 2016. Match-SRNN: Modeling the Recursive Matching Structure with Spatial RNN. *arXiv preprint arXiv:1604.04378* (2016).
[35] Mengqiu Wang and Luo Si. 2008. Discriminative probabilistic models for passage based retrieval. In *SIGIR*. ACM, 419–426.
[36] 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.
[37] Wensi Xi, Richard Xu-Rong, Christopher SG Khoo, and Ee-Peng Lim. 2001. Incorporating window-based passage-level evidence in document retrieval. *Journal of information science* 27, 2 (2001), 73–80.
[38] Jun Xu and Hang Li. 2007. Adarank: a boosting algorithm for information retrieval. In *SIGIR*. ACM, 391–398.
[39] Chengxiang Zhai and John Lafferty. 2001. A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval. In *SIGIR*. ACM, New York, NY, USA, 334–342. DOI:http://dx.doi.org/10.1145/383952.384019