A Multi-Granularity-Aware Aspect Learning Model for Multi-Aspect Dense Retrieval

Xiaojie Sun
CAS Key Lab of Network Data Science and Technology, ICT, CAS
University of Chinese Academy of Sciences
Beijing, China
sunxiaojie21s@ict.ac.cn

Keping Bi
CAS Key Lab of Network Data Science and Technology, ICT, CAS
University of Chinese Academy of Sciences
Beijing, China
bikeping@ict.ac.cn

Jiafeng Guo*
CAS Key Lab of Network Data Science and Technology, ICT, CAS
University of Chinese Academy of Sciences
Beijing, China
guojiafeng@ict.ac.cn

Sihui Yang
CAS Key Lab of Network Data Science and Technology, ICT, CAS
University of Chinese Academy of Sciences
Beijing, China
yangsihui22s@ict.ac.cn

Qishen Zhang
Zhongyi Liu
Guannan Zhang
Ant Group
Beijing, China
{qishen.zqs,zhongyi.lzy,zgn138592}@alibaba-inc.com

Xueqi Cheng
CAS Key Lab of Network Data Science and Technology, ICT, CAS
University of Chinese Academy of Sciences
Beijing, China
cxq@ict.ac.cn

ABSTRACT

Dense retrieval methods have been mostly focused on unstructured text and less attention has been drawn to structured data with various aspects, e.g., products with aspects such as category and brand. Recent work has proposed two approaches to incorporate the aspect information into item representations for effective retrieval by predicting the values associated with the item aspects. Despite their efficacy, they treat the values as isolated classes (e.g., "Smart Homes", "Home, Garden & Tools", and "Beauty & Health") and ignore their fine-grained semantic relation. Furthermore, they either enforce the learning of aspects into the CLS token, which could confuse it from its designated use for representing the entire content semantics, or learn extra aspect embeddings only with the value prediction objective, which could be insufficient especially when there are no annotated values for an item aspect.

Aware of these limitations, we propose a MUlti-granulaRity-aware Aspect Learning model (MURAL) for multi-aspect dense retrieval. It leverages aspect information across various granularities to capture both coarse and fine-grained semantic relations between values. Moreover, MURAL incorporates separate aspect embeddings as input to transformer encoders so that the masked language model objective can assist implicit aspect learning even without aspect-value annotations. Extensive experiments on two real-world datasets of products and mini-programs show that MURAL outperforms state-of-the-art baselines significantly. Code will be available at the URL[1].

[*] Jiafeng Guo is the corresponding author.
[1] https://github.com/sunxiaojie99/MURAL

WSDM '24, March 4–8, 2024, Merida, Mexico
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0371-3/24/03.
https://doi.org/10.1145/3616855.3635770

CCS CONCEPTS

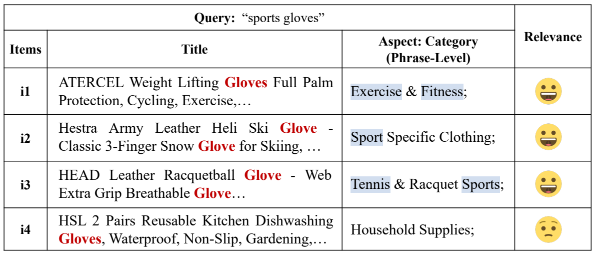
• **Information systems → Information retrieval**.

KEYWORDS

Dense Retrieval, Multi-Aspect, Pre-training

ACM Reference Format:
Xiaojie Sun, Keping Bi, Jiafeng Guo, Sihui Yang, Qishen Zhang, Zhongyi Liu, Guannan Zhang, and Xueqi Cheng. 2024. A Multi-Granularity-Aware Aspect Learning Model for Multi-Aspect Dense Retrieval. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining (WSDM '24), March 4–8, 2024, Merida, Mexico.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3616855.3635770

1 INTRODUCTION



Query: "sports gloves"			Relevance
Items	**Title**	**Aspect: Category (Phrase-Level)**	
i1	ATERCEL Weight Lifting **Gloves** Full Palm Protection, Cycling, Exercise,…	Exercise & Fitness;	😀
i2	Hestra Army Leather Heli Ski **Glove** - Classic 3-Finger Snow **Glove** for Skiing, …	Sport Specific Clothing;	😀
i3	HEAD Leather Racquetball **Glove** - Web Extra Grip Breathable **Glove**…	Tennis & Racquet Sports;	😀
i4	HSL 2 Pairs Reusable Kitchen Dishwashing **Gloves**, Waterproof, Non-Slip, Gardening,…	Household Supplies;	😕

Figure 1: An example of a query and its candidate items.

In recent years, dense retrieval methods have been extensively studied in both Information Retrieval (IR) and Natural Language Processing (NLP) communities [9]. On the shoulders of pre-trained language models (PLMs), they have achieved compelling performance. However, they are mostly studied for unstructured data and have not investigated how to effectively leverage the aspect information of structured data, such as category for products and affiliation for people. For example, in Figure 1, the query "sports gloves" targets gloves for sports use so kitchen gloves should be avoided. It is obvious that the category of the four items could help to differentiate various types of gloves and improve retrieval performance. Unfortunately, it remains largely unexplored to effectively leverage such aspect information in dense retrieval.

Recently, Kong et al. [13] has proposed two effective models for multi-aspect dense retrieval, i.e., MTBERT and MADRAL. These methods follow a typical paradigm of learning aspect embeddings with an auxiliary objective of predicting their associated values [2, 3]. A concrete example is that the embedding of aspect "category" for i4 in Figure 1 will be learned by predicting its value, i.e., "Household Supplies". Although effective, they consider the values of an aspect as isolated classes and neglect the potential correlation between various values, which could result in sub-optimal performance. Taking the items in Figure 1 for instance, although they fall into four separate categories, the first three are relevant to the user query "sports gloves" while the last is not. The auxiliary objective of predicting their categorical IDs treats each category equally and may not capture their fine-grained relations.

Noticing this issue, we propose to leverage the aspect information at even finer granularities, such as the word and token levels, in addition to the previously considered phrase-level granularity. Then, for the items in Figure 1, when we break their category phrases into small pieces, the relation between the first three will be clearer since they all have sports-related descriptions such as exercise, sport, tennis, etc. Moreover, from a linguistic perspective, coarser granularities such as sentences and phrases convey more specific information while finer units usually carry more general information [22]. Since different granularities could express various levels of intent, we incorporate multiple granularities of aspect annotation prediction to assist query/item representation learning.

Our model is named MURAL, short for a MUlti-granulaRity-aware Aspect Learning model. It incorporates separate aspect embeddings before the content tokens and after CLS as inputs to the transformer layers (shown in Figure 3). Then, on the top layer, the aspect embeddings are supervised with value predictions at various levels of granularities (e.g., phrase, word, and token). MURAL has several advantages over state-of-the-art methods, i.e., MTBERT and MADRAL (See Figure 2): First, in contrast to MTBERT which mixes the information from item aspects and the overall content semantics in CLS, MURAL represents the two types of information separately and allows for more interactions between them with a gating mechanism. Second, in contrast to MADRAL which only learns the aspect embeddings with the value prediction objective during pre-training, MURAL also guides the aspect embeddings to learn from the masked language model loss. This could assist implicit aspect learning even when there are no annotated values for an item aspect. Last and most importantly, by incorporating the aspect information across various granularities, MURAL could capture the semantic relations between the aspect values at different levels, contributing more to the retrieval performance.

We conduct extensive experiments on two real-world search datasets with rich aspect information. Experimental results show that our method outperforms competitive baselines significantly on both datasets. It is remarkable that our model achieves compelling performance even without the supervision of aspect annotations, which means that useful implicit representations can be learned by MURAL even when the aspect information is not used. Ablation studies on different granularities show that each granularity can contribute to the multi-aspect retrieval performance and combining them all lead to much better results.

2 RELATED WORK

Dense Retrieval. Dense retrieval models typically use a bi-encoder structure for independent query and item encoding, with relevance measured through a simple similarity function (such as dot product). Karpukhin et al. [10] initializes the encoder with BERT and combines it with in-batch negatives, achieving better performance than early models. After that, researchers began to explore various fine-tuning techniques to train a better dense retriever, including hard negative mining [23, 31], knowledge distillation [28], and multi-vector representation [11, 19, 33]. For example, Xiong et al. [31] proposed to dynamically mine hard negatives during training by periodically refreshing the index. Luan et al. [19] captures information of items from different perspectives by using the first k document token embeddings as the item representation. Based on this, Zhang et al. [33] added k special tokens before the item input to obtain the multi-vector representation. These multi-vector methods aim to extract multiple underlying semantic information from the item. In contrast, our method explicitly considers explicit multi-aspect information modeling. Additionally, our method outputs only a single representation vector for each item, saving space and time for indexing items.

Recently, Kong et al. [13] introduced two methods for incorporating explicit aspect information into a single representation vector. The first method employs CLS embeddings to simultaneously perform aspect classification tasks for multiple aspects. The second method adds an attention network to the PLM, enabling it to separately model multiple aspects, followed by aspect fusion. Their differences with our method will be introduced in Section 4.

Multi-Field Retrieval. The effective utilization of multi-field information (*e.g.,* title, keyword, description) in documents has been studied for long. Before PLM appears, many neural ranking models were proposed to effectively leverage item structure [4, 17, 32]. For example, Zamani et al. [32] aggregated field-level representations to obtain item representations and employed a matching network for final relevance score prediction. In the PLM era, research has continuously focused on the utilization of multi-field information [26, 27]. For example, Shan et al. [26] proposed the field-level local matching loss, calculated based on the query and each document field representation. Sun et al. [27] treated aspect as text and proposed an effective pre-training method to capture the bi-directional interactions between aspect and content texts. The difference between multi-aspect and multi-field is that fields contain an infinite textual value space, usually composed of variable-length unstructured text. Conversely, an aspect has a defined set of finite values, acting as "labels" for structured items. Given this, they face different core challenges, and effectively utilizing multiple aspects' information is a valuable research direction.

Pre-trained Bi-encoder. Researchers have explored pre-training models for retrieval with the bi-encoder architecture[6, 7, 14, 18, 20, 30]. For example, Gao and Callan [6] added extra head layers atop the Transformer, with shortcut connections between early outputs and the head, enhancing the CLS embedding of the encoder. Lu et al. [18] pre-trained an auto-encoder with a weak decoder

Table 1: A summary of main notations used in this paper.

Notation	Meaning
$X = (x_1, x_2, ..., x_n)$	The input token sequence of query/item.
$\mathbf{h}_X$	The final representation for the input X.
$A = \{a_i\}, i = 1, ..., \|A\|$	A set of aspects, *e.g.*, $\{brand, color, category\}$
$G = \{g_i\}, i = 1, ..., \|G\|$	The set of language granularities, *e.g.*, $\{phrase, word, token\}$
$V_{a_i}^{g_j}$ or V_a^g, $i = 1, ..., \|A\|, j = 1, ..., \|G\|$	The aspect value vocabulary of aspect a_i at granularity level g_j, *e.g.*, $V_{category}^{word} = \{shoes, clothing, ...\}$
$\mathbf{h}_{a_i}^{g_j}$ or $\mathbf{h}_a^g$, $i = 1, ..., \|A\|, j = 1, ..., \|G\|$	The aspect embedding for query's or item's aspect a at granularity g.
$E_{a_i}^{g_j}$ or E_a^g, $i = 1, ..., \|A\|, j = 1, ..., \|G\|$	The aspect value embedding table for aspect a at granularity g.
$\mathcal{A}_{a_i}^{g_j}$ or $\mathcal{A}_a^g$, $i = 1, ..., \|A\|, j = 1, ..., \|G\|$	The set of aspect value annotations for query's or item's aspect a at granularity g, $\mathcal{A}_a^g \subset V_a^g$

for document representation learning. Differing from these pre-training methods targeted at unstructured data, we investigate how to infuse explicit aspect information into the encoder representation during pre-training. In the future, we will explore how to integrate our approach with existing research.

3 PRELIMINARIES

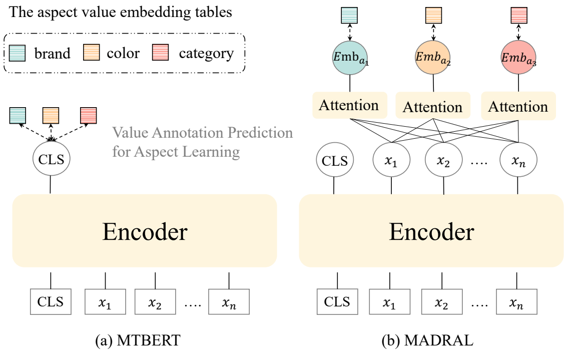
Dual Encoding. The standard PLMs, *e.g.*, BERT [5], take a token sequence $X = (x_1, ..., x_n)$ as input, and generate contextualized representations as:

$$\mathbf{h}(x_0), \mathbf{h}(x_1)...\mathbf{h}(x_n) = \Phi_{enc}(X), \ \mathbf{h}(x_i) \in \mathbb{R}^H, \quad (1)$$

where H denotes the hidden size, and $x_0 = [CLS]$ is a special token added to the beginning. The representation $\mathbf{h}(x_0)$ is commonly used as the final representation for the input X. In dense retrieval, the bi-encoder architecture is widely adopted, where the query Q and item I are separately encoded using the PLM to obtain their respective representation vectors[16]. Then, a simple scoring function is used to calculate the similarity between these two vectors.

Aspect Learning. In dense retrieval, aspect learning involves using aspect information to enhance retrieval performance when queries or items are associated with varying aspects (e.g., *brand*, *color*, *category* in product search). In addition to the content text $X = (x_1, ..., x_n)$ (e.g., query, item title), a query or item can be associated with multiple aspects, and we denote the set of these aspects as $A = \{a_i\}_{i=1}^{|A|}$. For simplicity, when the context is clear, we omit the subscript i in a_i. For each aspect a, there exists a finite vocabulary of aspect values, represented as V_a, along with a corresponding embedding table $E_a \in \mathbb{R}^{|V_a| \times H}$ that contains embeddings for each value of aspect a. Figure 2 shows the aspect learning in two state-of-the-art multi-aspect dense retrievers [13]. Both approaches utilize content text as the encoder input. Specifically, MTBERT reuses CLS to represent aspects, whereas MADRAL constructs embeddings for the $|A|$ aspects by attending to the final layer of content tokens. Both methods train the aspect embeddings by predicting the corresponding value annotation ID in V_a for each of the $|A|$ aspects.

Multi-Granularity. Different granularities of text strings capture semantic information at varied levels. Coarse grains such as sentences or phrases often express more specific intent than finer grains like words or tokens. Therefore, relying solely on phrase-level value prediction, as previous methods do, might not yield effective aspect representations. For example, if a product category value is "handmade products", its word-level granularity values would be "$[handmade, products]$", and its token-level granularity values would be "$[hand, \#\#made, products]$". Formally, we denote the set of granularities as G, where each g (with $g \in G$) represents a specific granularity. In this paper, we use three granularities: $G = \{phrase, word, token\}$. We use V_a^g to represent the value vocabulary obtained by decomposing aspect a's values at granularity g. The corresponding aspect value embedding table becomes $E_a^g \in \mathbb{R}^{|V_a^g| \times H}$. We list the frequently used notations in Table 1.



(a) MTBERT (b) MADRAL

Figure 2: SOTA multi-aspect dense retrieval models.

4 METHODOLOGY

In this section, we propose a MUlti-granulaRity-aware Aspect Learning model (MURAL) for multi-aspect dense retrieval and introduce its core components. As in [13], MURAL is also based on BERT [5]. Since MURAL encodes both items and queries in the same way, we only use items for illustration.

4.1 Aspect Representations

It is crucial to represent aspects reasonably in a pre-trained model so that aspect learning can guide their training effectively. To fully exploit the capabilities of the Transformer encoders, as shown in Figure 3, we introduce several tokens after CLS and before the content tokens to represent aspects from various perspectives. This aligns with the way CLS is obtained and these tokens can interact with the content tokens sufficiently. During pre-training, these inserted embeddings can act as different views of context when predicting the masked tokens in the content. In this way, these embeddings can also learn from the masked language model objective and capture the content semantics from various implicit views, which could bring more benefits, especially when there are no value annotations for an aspect.

Comparison with Previous Methods. As shown in Figure 2, MTBERT[13] reuses the CLS token to predict the values of item aspects, which enforces CLS to mix the information from item aspects with the overall content semantics it is originally designated to capture. The balance between the two cannot be automatically learned and CLS could be confused about what it should learn. MADRAL [13] represents each item aspect separately by attending to the final representations of content tokens and learns the aspect embeddings by predicting their associated values. During pre-training, the only guidance for the aspect embeddings is this value prediction

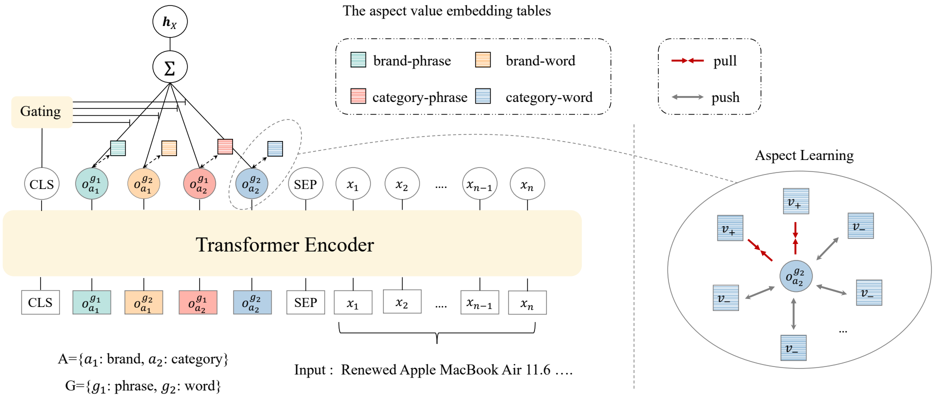


Figure 3: Our MURAL with single-objective-based grouping in a simplistic scenario of two aspects and two granularities.

objective, which could be insufficient to learn them well. What is worse, they will not be updated when there are no aspect-value annotations. In contrast, in MURAL, the aspect information would not mix with the overall content semantics in CLS. With the gating mechanism, they can interact more and the aspect importance can be learned automatically. Moreover, the aspect embeddings can be guided by the masked language model loss as well, which not only benefits their representation learning but also facilitates implicit aspect learning without aspect-value annotations.

4.2 Aspect Learning

For simplicity, we use an example of aspect a at granularity g to illustrate. For aspect learning in MURAL, there are two important components: value representation and the aspect learning objective.

Value Representation. To conduct effective aspect learning by predicting the value annotations of an aspect in terms of both coarse and fine grains, value representations play an important role. There are two options: 1) Sharing the existing token embeddings in the backbone PLM and calculating the value embeddings of word-level and phrase-level grains by a projection function. It can reuse the semantic information carried in the PLM tokens. However, the token embeddings are learned towards the goals of both PLM and aspect learning, which may interfere with each other. 2) Declaring separate value embedding tables, which is consistent with [13] and the research before the PLMs era in [3]. The extra value embeddings could serve the model to conduct aspect learning better without other interventions. However, if trained from scratch, these new parameters may be difficult to optimize. We refer to these two options as "shared" and "unshared" respectively in terms of whether to share the underlying token embeddings with the existing encoder. We investigate both ways in our experiments (see Section 6.2).

Specifically, for the "shared" option, we first tokenize each aspect value v in V_a^g using the BERT tokenizer. Then, we extract their embeddings from BERT's embedding tables and use a projector function on the token embeddings to obtain the corresponding value embedding e_v during training. In this paper, we adopt average pooling as the projection function since it is simple and produces similar results to other methods in our preliminary experiments.

For the "unshared" option, each aspect a has a separate embedding table E_a^g for each granularity g, storing the embeddings of its values V_a^g. Instead of training these tables from scratch, we initialize the tables using the average token embeddings in the PLM for each value v at granularity g (the same as the initial embedding in the "shared" option). This gives the new embeddings a decent starting point in the semantic space and has the freedom to better conduct aspect learning, the benefit of which will be shown in Section 6.2.

Aspect Learning Objective. Once we obtain the representation of aspect a at granularity g, denoted as $\mathbf{h}_a^g(X) \in \mathbb{R}^H$, we adopt the widely-used group-wise contrastive loss to pre-train the encoder. It aims to bring the source representation closer to instances in its target group while distancing it from representations of other groups [12].

$$\mathcal{L}_a^g(X) = -\frac{1}{|\mathcal{A}_a^g|} \sum_{v^+ \in \mathcal{A}_a^g} log \frac{exp(sim(\mathbf{h}_a^g(X), \mathbf{e}_{v^+}))}{\sum_{v \in V_a^g} exp(sim(\mathbf{h}_a^g(X), \mathbf{e}_v))}, \tag{2}$$

where $\mathbf{e}_v/\mathbf{e}_{v^+} \in \mathbb{R}^H$ is the aspect value embedding from E_a^g, $sim(\cdot)$ is the dot-product function, and $\mathcal{A}_a^g$ is the set of aspect value annotations for aspect a at granularity g.

4.3 Multi-Granularity-Aspect Grouping

Assume there are $|A|$ aspects and $|G|$ granularities, our goal is to facilitate aspect learning for each aspect of each granularity, totally |A|*|G| learning objectives. A straightforward approach is to use a single representation to handle these multiple objectives. However, this method enforces all the information to be compressed together, severely limiting the learning capacity of each objective. Therefore, we introduce three grouping schemes to integrate multi-granularities and multi-aspects: Single-objective-based Grouping, Granularity-based Grouping, and Aspect-based Grouping.

Single-objective-based Grouping. As shown in Figure 3, when there are only a few aspects and granularities, we can directly introduce $|A| * |G|$ tokens in the input sequence X to capture the item semantics from $|A| * |G|$ views. Each of them accounts for a single objective among the multi-granularity-aspect combinations. Specifically, we obtain a sequence of $X = (x_0, o_1, ..., o_{|A|*|G|}, x_1, .., x_n)$. We utilize the hidden vector $\mathbf{h}(o_k)(k = 1, ..., i * j)$ from the final layer as the item representation from the perspective of aspect a_i at the granularity g_j. The aspect learning loss function becomes:

$$\mathcal{L}_{\mathcal{A}}(X) = \frac{1}{|A| * |G|} \sum_{i=1}^{|A|} \sum_{j=1}^{|G|} \mathcal{L}_{a_i}^{g_j}(X). \tag{3}$$

When $|A| * |G|$ is large, adding a significant number of tokens can adversely affect the semantic representation of the original input.

Hence, it becomes essential to further group the objectives across various granularities and aspects.

Granularity-based Grouping. The same granularity indicates the same level of semantic information, and grouping the objectives at the same grain is a reasonable option. In this case, $|G|$ tokens will be inserted into the input sequence, yielding $X = (x_0, o_1, ..., o_{|G|}, x_1, x_2, .., x_n)$. Their encoded representations become $\mathbf{h}(o_j)$ $(j = 1, ..., |G|)$, representing the item from the perspective of all the aspect information at granularity g_j. A single aspect embedding accounts for the loss $\mathcal{L}_{a_i}^{g_j}(X)$ of all the aspects a_i $(i = 1, ..., |A|)$ of granularity g_j, i.e., $\mathbf{h}_{a_i}^{g_j}$ $(i = 1, ..., |A|)$ are the same for granularity g_j. The aspect learning objective is:

$$\mathcal{L}_{\mathcal{A}}(X) = \frac{1}{|G|}\sum_{j=1}^{|G|}\mathcal{L}_{g_j}(X), \text{where } \mathcal{L}_{g_j}(X) = \frac{1}{|A|}\sum_{i=1}^{|A|}\mathcal{L}_{a_i}^{g_j}(X). \tag{4}$$

Aspect-based Grouping. An alternative option is to group the objectives across multi-granularity-aspects by aspects so that different aspect information will not mix together and various levels of granularities could benefit each other. Here, we introduce $|A|$ guiding tokens before the content tokens: $X = (x_0, o_1, ..., o_{|A|}, x_1, x_2, .., x_n)$. The hidden vector $\mathbf{h}(o_i)$ $(i = 1, ..., |A|)$ captures the representations of all granularities for the input item corresponding to aspect a_i. In particular, when calculating the loss $\mathcal{L}_{a_i}^{g_j}(X)$ using equation 2, the representation $\mathbf{h}_{a_i}^{g_j}$ of aspect a_i remains consistent across different granularities. Under this aggregation method, loss $\mathcal{L}_{\mathcal{A}}$ can be reformulated as follows:

$$\mathcal{L}_{\mathcal{A}}(X) = \frac{1}{|A|}\sum_{i=1}^{|A|}\mathcal{L}_{a_i}(X), \text{where } \mathcal{L}_{a_i}(X) = \frac{1}{|G|}\sum_{j=1}^{|G|}\mathcal{L}_{a_i}^{g_j}(X). \tag{5}$$

Grouping by granularities or aspects reduces the number of guiding tokens, accommodating scenarios with numerous aspects and granularities. Their model architectures stay the same as Figure 3, except that aspect learning objectives for the same granularities or aspects are conducted on the shared token.

4.4 Aspect Embedding Fusion

For efficiency concerns, it is necessary to consolidate multiple embeddings into a single one to minimize storage and computation costs. Inspired by [13], we adopt the "CLS-Gating" fusion mechanism in MURAL. To illustrate the fusion process, we present an example using the single-objective-based grouping approach discussed in Section 4.3. Specifically, we pass the CLS embedding through a linear layer and a softmax function to compute the weighting scores for $\mathbf{h}(o_1), ..., \mathbf{h}(o_K)$, where $K = |A| * |G|$:

$$\mathbf{w} = Softmax(U\mathbf{h}(x_0) + b) \in \mathbb{R}^K, \tag{6}$$

where $U \in \mathbb{R}^{K \times H}$ and $b \in \mathbb{R}^K$ are trainable parameters. Then, we utilize the learned weights to fuse multiple embeddings, thereby obtaining the final encoded representation of the input X:

$$\mathbf{h}_X = \sum_{k=1}^{K} w_k \cdot \mathbf{h}(o_k). \tag{7}$$

Table 2: Aspect-Related Dataset Statistics. It presents the percentage of queries/items with non-empty aspect values in the pre-training corpus and the aspect value vocabulary sizes at various granularities: phrase, word, and token.

	MA-Amazon	*Alipay*
aspect	*item*	*item / query*
brand	94% (5k,6k,5k)	0.6%/44% (9k,11k,3k)
color	67% (2k,1k,1k)	–
category	87% (8k,5k,5k)	90%/91% (457,650,548)

4.5 Training Objectives

Pre-training. As discussed in previous work [21], the Masked Language Model (MLM) [29] task could help construct good text representation for IR. Therefore, similar to [13], we also adopt MLM as one of the pre-training objectives besides aspect learning.

$$\mathcal{L}_{MLM}(X) = -\sum_{w \in masked(X)} logP(w|X_{\backslash masked(X)}), \tag{8}$$

where X means the input sentence, $masked(X)$ and $X_{\backslash masked(X)}$ denotes the masked tokens and the remaining tokens from X, respectively.

We then pre-train the Transformer encoder using the aspect learning loss jointly with the MLM loss, as follows,

$$\mathcal{L}_{total}(X) = \mathcal{L}_{MLM}(X) + \lambda\mathcal{L}_{\mathcal{A}}(X), \tag{9}$$

where λ is the hyperparameter.

Fine-Tuning. We adopt the following in-batch softmax cross entropy loss $\mathcal{L}_{SCE}$ as the learning objective during fine-tuning. Note that although the aspect learning loss could also be added during fine-tuning, our experimental results show no significant improvements for all the multi-aspect retrievers. Hence, we omit this objective in this paper.

$$\mathcal{L}_{SCE} = -log\frac{exp(sim(\mathbf{h}_Q, \mathbf{h}_{I^+}))}{exp(sim(\mathbf{h}_Q, \mathbf{h}_{I^+})) + \sum_{I^-} exp(sim(\mathbf{h}_Q, \mathbf{h}_{I^-}))}. \tag{10}$$

5 EXPERIMENTAL SETTINGS

5.1 Datasets

We use the following two large-scale search datasets from real-world platforms with rich aspect information for our experiments. The aspect-related statistics of the two datasets are in Table 2. (1) **Multi-Aspect Amazon ESCI Dataset (MA-Amazon).** MA-Amazon [27] enriches the English portion of the Amazon ESCI [24] dataset with item category information. In MA-Amazon, only items have annotations for *brand*, *color* and *category*. The item corpus contains 482K distinct products, which are used for pre-training. The retrieval dataset has 17k, 3.5k, and 8.9k queries for fine-tuning, validation, and testing respectively, without any query overlaps. For each query, the retrieval dataset provides 20.1 items on average, along with their ESCI relevance judgments (*Exact*, *Substitute*, *Complement*, *Irrelevant*), indicating each item's relevance to the given query. Following [24], we treat *Exact* as positive and all others as negatives for fine-tuning and metrics requiring binary labels. (2) **Alipay Search Dataset.** Alipay is a Chinese mini-program (app-like service) search dataset. In Alipay, both queries and items are annotated with two aspects: *brand* and *category*. We conduct pre-training on both a query corpus, containing 1.3M unique queries, and an item corpus with 1.8M distinct items. The retrieval dataset contains 60k, 3.3k, and 3.3k real user queries for fine-tuning, validation, and testing respectively, without query overlaps. Note that

Table 3: Comparisons between MURAL and the baselines. The best results (excluding MURAL-CONCAT) are in bold. †, ‡, and ∗ indicate significant improvements over the best baselines in the first/second group and the backbone BIBERT, respectively.

Method	MA-Amazon			Alipay		
	R@100	R@500	NDCG@50	R@100	R@500	NDCG@50
BIBERT	0.6075	0.7795	0.3929	0.4464	0.6284	0.2033
Condenser	0.6091	0.7801	0.3960	0.4520	0.6423	0.2072
BIBERT-CONCAT	0.6137	0.7814	0.4005	0.4517	0.6291	0.2103
MTBERT	0.6137	0.7852	0.3969	0.4498	0.6280	0.2064
MADRAL	0.6088	0.7815	0.3950	0.4506	0.6383	0.2057
MUR	$0.6282^{†‡*}$	$0.7943^{†‡*}$	$0.4151^{†‡*}$	0.4556^{*}	$0.6458^{‡*}$	0.2046
MURAL	$\mathbf{0.6371}^{†‡*}$	$\mathbf{0.8023}^{†‡*}$	$\mathbf{0.4228}^{†‡*}$	$\mathbf{0.4630}^{†‡*}$	$\mathbf{0.6519}^{†‡*}$	$\mathbf{0.2177}^{†‡*}$
MURAL-CONCAT	$0.6389^{†‡*}$	$0.8005^{†‡*}$	$0.4281^{†‡*}$	$0.4669^{†‡*}$	$0.6474^{‡*}$	$0.2124^{†*}$

the queries for validation and testing do not appear in the pre-training query corpus. Each instance in the relevance dataset is a <*query*, *item*, *label*> triplet, where the label indicates the manually annotated binary relevance of this query-item pair.

5.2 Baselines

We adopt the following dense retrieval baselines for comparison, including models using or without using aspect information: (1) **BIBERT** [15, 25]: A standard bi-encoder baseline and the backbone of MURAL, using CLS encoding of the BERT-based encoder for both query and item representations. BIBERT is pre-trained with MLM loss and fine-tuned with loss $\mathcal{L}_{SCE}$ (Equation 10). (2) **Condenser**[6]: A pre-trained method tailored for unstructured textual dense retrieval. It introduces a short circuit between middle-layer tokens (excluding CLS) and their corresponding head-layer tokens during pre-training, optimizing the CLS embedding to encapsulate more information. (3) **BIBERT-CONCAT**: It treats the aspect values as texts and concatenates them with the query/item content during pre-training with MLM. During fine-tuning, since the concatenation with query could change query semantics , we only concatenate item aspects for relevance matching. (4) **MTBERT**[13]: A multi-task (MT) learning model based on BIBERT. Besides MLM during pre-training, it conducts $|A|$ aspect prediction tasks using CLS. (5) **MADRAL**[13]: It incorporates an aspect extraction attention network to extract $|A|$ aspect representations for both queries and items. These embeddings are learned from aspect prediction tasks during pre-training and fused to yield the final representation during fine-tuning. (6) **MURAL, MUR and MURAL-CONCAT**: MURAL is our proposed multi-aspect dense retrieval model. In contrast, MUR disables aspect learning. Specifically, when λ in Equation 9 is set to 0, MURAL regresses to MUR. MURAL-CONCAT employs the same aspect-content text concatenation strategy as BIBERT-CONCAT for the model input. Note that unless the model name includes "-CONCAT", the model input consists solely of content text.

5.3 Implementation Details

We implemented MURAL and all the baselines by ourselves to ensure consistent implementation details and fair comparisons.

5.3.1 Pre-training. For all methods, the encoder is shared for both queries and items to facilitate knowledge sharing. Specifically, we pre-train on a corpus consisting of the item corpus or a mixture of the query and item corpus (when query aspect annotations are available) to obtain the shared encoder for fine-tuning.

Muti-granularity Value Collection. Given an aspect a and its original aspect vocabulary at the phrase level, we obtain its word and token granularity vocabularies: For word granularity, we segment each aspect value v by spaces and punctuation (for English) or employ the Jieba tool [1] (for Chinese), and eliminate duplicates to aggregate the generated "words". For token granularity, we merge the token list obtained by processing each aspect value v with the BERT tokenizer to create the corresponding vocabulary set.

Model Pre-training. We initialize all the BERT components using Google's public checkpoint and employ the Adam optimizer with the linear warm-up technique. The learning rate and epoch for the MA-Amazon/Alipay dataset are set to 1e-4/5e-5 and 20/10, respectively. The maximum token length is 156, the MLM mask ratios are 0.15 for items and 0.3 for queries. For all methods requiring adjustment of training objective scaling coefficients, we uniformly select coefficients based on their validation set performance after fine-tuning. These coefficients vary from 0.1 to 1, in 0.1 intervals. For our method, we set λ in Eq.9 to 0.1. We use the following fine-tuning procedures to evaluate pre-trained model checkpoints every two epochs and select the best one on the validation set.

5.3.2 Fine-tuning. For both datasets, we fine-tune all the models for 20 epochs with Tevatron toolkit[8]. Following the previous work [10], we include a hard negative sample for each query besides in-batch negatives. We use a learning rate of 5e-6 and a batch size of 64. The maximum token lengths are set at 32 for queries and 156 for items. All the models are trained with relevance loss $\mathcal{L}_{SCE}$ (Eq.10).

5.3.3 Evaluation Metrics. We report R@100, R@500, and NDCG@50. Following [24], we assign the gains of 1.0, 0.1, 0.01, and 0.0 to E, S, C, and I, respectively, for MA-Amazon. We conduct two-tailed paired t-tests ($p < 0.05$) to identify significant differences.

6 EXPERIMENT RESULTS

6.1 Overall Performance

We compare MURAL with baseline models both utilizing and without utilizing aspect information. As shown in Table 3, we have the following observations: (1) The models that leverage the aspect information (methods except for BIBERT, Condenser, and MUR) outperform their backbone (BERT) without using it. Among the multi-aspect dense retrievers, MURAL performs the best with a significantly large margin. This confirms the necessity of incorporating aspects in query/item representation learning. (2) On MA-Amazon, MADRAL underperforms the simpler MTBERT. We believe this is due to the less pre-training data of MA-Amazon and the absence

Table 4: Variants of MURAL on MA-Amazon. † indicates significant differences from the best option.

Method	R@100	R@500	NDCG@50
MURAL $^{unshared\_single}$	$0.6336^{\dagger}$	0.8005	$0.4195^{\dagger}$
MURAL $^{unshared\_granu}$	**0.6371**	**0.8023**	**0.4228**
MURAL $^{unshared\_aspect}$	$0.6340^{\dagger}$	$0.8003^{\dagger}$	$0.4195^{\dagger}$
MURAL $^{shared\_single}$	$0.6300^{\dagger}$	$0.7980^{\dagger}$	$0.4171^{\dagger}$
MURAL $^{shared\_granu}$	$0.6333^{\dagger}$	$0.7996^{\dagger}$	$0.4184^{\dagger}$
MURAL $^{shared\_aspect}$	$0.6321^{\dagger}$	$0.7995^{\dagger}$	$0.4173^{\dagger}$
MURAL $^{unshared\_randinit}$	$0.6337^{\dagger}$	0.8006	$0.4196^{\dagger}$
MURAL $^{first\_k}$	$0.6141^{\dagger}$	$0.7873^{\dagger}$	$0.4009^{\dagger}$
MURAL $^{no\_cls\_gating}$	$0.6212^{\dagger}$	$0.7890^{\dagger}$	$0.4064^{\dagger}$

of aspect annotations for queries, which makes the aspect embeddings of MADRAL not sufficiently learned. In contrast, MURAL achieves compelling performance consistently. (3) Condenser, a more advanced pre-trained model for unstructured text retrieval, sometimes outperforms the baseline multi-aspect dense retrievers. Notably, the gains from advanced pre-training techniques are orthogonal to our method. Our approach can be easily incorporated into stronger backbones like Condenser and could achieve even better performance. We leave it in our future work. (4) BIBERT-CONCAT performs better than MTBERT and MADRAL in terms of some metrics on the two datasets, indicating that concatenating aspects as text strings can be beneficial. However, query aspects should be taken special care of during relevance matching in order to achieve good performance. Incorporating both concatenation and aspect prediction in the same model (MURAL-CONCAT) does not always result in better performance than without concatenation. We have similar observations with MTBERT and MADRAL, but due to the space concern, we do not report them. The reason may be that the model learns unwanted shortcuts when using the aspect both as the model input and the learning objective. (5) Our method shows competitive performance even without using aspect annotations (MUR). MUR outperforms most baseline models in terms of all the metrics except NDCG@50 on Alipay. This indicates that MUR can capture complementary information from implicit perspectives for the final representation. It also confirms the advantages of aspect representations and the MLM training for the aspects in MURAL.

6.2 Studies on Model Variants

In this subsection, we study various options for the essential components in MURAL. For reproducibility, all experiments are conducted on the public MA-Amazon dataset.

Studies on Value Representations. In Table 4, we observe that using an independent value embedding space (the "unshared" option in Section 4.2) leads to better performance. As we mentioned earlier, the "shared" option optimizes token embeddings both towards the objectives in BERT and the aspect learning, which could interfere with each other and limit the capacity of the model on aspect prediction. However, under the "unshared" option, it may be difficult to optimize the separate value embeddings from scratch while other parameters only need fine-tuning. To see whether this affects model performance, instead of using the same initial state as the "shared" option, we randomly initialize the embeddings while keeping other best settings in MURAL. The harmed performance of MURAL $^{unshared\_randinit}$ in Table 4 confirms our presumption and shows the benefit of decent initialization states.

Table 5: Ablation studies of MURAL in terms of category and granularity on the MA-Amazon dataset. †, ‡ indicate significant differences over MURAL and BIBERT.

Method	R@100	R@500	NDCG@50
BIBERT	0.6075	0.7795	0.3929
MURAL	**0.6371**	**0.8023**	**0.4228**
MURAL $^{only\ brand}$	$0.6289^{\dagger\ddagger}$	$0.7951^{\dagger\ddagger}$	$0.4168^{\dagger\ddagger}$
MURAL $^{only\ color}$	$0.6284^{\dagger\ddagger}$	$0.7942^{\dagger\ddagger}$	$0.4158^{\dagger\ddagger}$
MURAL $^{only\ category}$	$0.6315^{\dagger\ddagger}$	$0.7994^{\dagger\ddagger}$	$0.4166^{\dagger\ddagger}$
MURAL $^{only\ phrase}$	$0.6315^{\dagger\ddagger}$	$0.7983^{\dagger\ddagger}$	$0.4185^{\dagger\ddagger}$
MURAL $^{only\ word}$	$0.6309^{\dagger\ddagger}$	$0.7994^{\dagger\ddagger}$	$0.4194^{\dagger\ddagger}$
MURAL $^{only\ token}$	$0.6305^{\dagger\ddagger}$	$0.7982^{\dagger\ddagger}$	$0.4192^{\dagger\ddagger}$

Studies on Grouping Methods. In Table 4, we observe that MURAL $^{unshared\_granu}$, which groups objectives by granularity, performs the best. Note that on the Alipay dataset, which has fewer aspects, MURAL $^{unshared\_single}$, single-objective-based grouping, has the best performance. This is consistent with our claim in Section 4.3 that as the aspect count increases, further grouping benefits model training. Based on these observations, we suggest: (1) For small numbers of aspects and granularities, simply use independent learning for each objective (MURAL $^{unshared\_single}$). (2) When there are more aspects and granularities, grouping multiple objectives in one guiding token can be a better choice.

Studies on Guiding Tokens and Fusion Methods. Instead of adding separate guiding tokens for aspect learning, we study a variant that reuses the same amount of tokens at the beginning of the input sequence to conduct aspect learning, denoted as MURAL $^{first\_k}$. The results show that MURAL $^{first\_k}$ has similar or better performance to the best baseline in Table 3 but is significantly worse than the best variant of MURAL. This indicates that the multi-granularity-aware aspect learning is beneficial but using separate guiding tokens to conduct the learning is needed.

To study whether CLS-Gating (introduced in Section 4.4) is helpful for aspect embedding fusion, in MURAL $^{no\_cls\_gating}$, we remove it and use the CLS embedding as the final representation. CLS naturally fuses the aspect embeddings in the second-to-last layer while the aspect learning is conducted in the last layer. This variant performs better than the best baseline in Table 3 but is worse than the best variant. It indicates that the fusion should be carried on the final aspect embeddings with a proper weighting mechanism.

6.3 Ablation Studies

We ablate various components of multi-aspect, multi-granularity, and query/item aspect learning. In this section, our experiments are also based on the enriched MA-Amazon dataset. Additionally, we validate the importance of the query and item side effects on the Alipay dataset, since MA-Amazon lacks query-side aspect information.

Effect of Aspects and Granularities. In Table 5, we first study the impact of multi-aspect and multi-granularity in MURAL. We find that: (1) Every aspect contributes to the model performance, especially *category*, consistent with [13]. (2) Each granularity alone

Table 6: Ablation of query and item aspects on Alipay. †, ‡ indicate significant differences over MURAL and BIBERT.

Method	R@100	R@500	NDCG@50
BIBERT	0.4464	0.6284	0.2033
MURAL	**0.4630**	**0.6519**	**0.2177**
MURAL $^{-query}$	0.4569‡	0.6400†‡	0.2126†‡
MURAL $^{-doc}$	0.4573‡	0.6454†‡	0.2103†‡

Table 7: The category aspect accuracy on Alipay dataset.

Method		query		doc	
		pre-train	fine-tune	pre-train	fine-tune
MTBERT	phrase	0.86	0.11	0.97	0.20
MADRAL	phrase	0.88	0.81	0.98	0.87
MURAL	phrase	0.89	0.85	0.98	0.93
	word	0.88	0.82	0.97	0.93
	token	0.89	0.73	0.97	0.80

is conducive to model performance and combining them all leads to even better results. Different granularities capture semantics at distinct levels and they can complement each other.

Effect of Query/Item Aspects. We disable the aspect learning on the query/item side in Table 6. We observe that both query and item aspects are beneficial and query aspects have a larger impact, which is also consistent with [13]. This is not surprising since query aspects are obtained from query analysis such as intent classification and carry more additional information.

6.4 Aspect Learning Accuracy

In Table 7, we compare the accuracy of MURAL with baseline methods after pre-training and fine-tuning to understand the aspect learning process better. We only analyze the most important aspect - *category* on Alipay. MA-Amazon and other aspects have similar conclusions. Considering that each item may have multiple category annotations, we use Accuracy@3 to calculate accuracy. Evaluation of the query and item aspect prediction is based on the test query set and item corpus of the Alipay dataset, respectively.

First, all methods have high accuracy after aspect learning in pre-training while lower accuracy after fine-tuning. Since we only use relevance loss during fine-tuning, it is expected that the accuracy will drop. In our experiments, we find that adding aspect learning loss during fine-tuning enhances aspect prediction accuracy but will harm retrieval performance. We speculate that this objective guides the model parameters to somewhere not aligned with the relevance-matching objective. Hence, higher aspect prediction accuracy does not always co-occur with better retrieval performance.

Secondly, the prediction accuracy of MTBERT drops dramatically after fine-tuning. Since MTBERT uses the same CLS token to conduct relevance matching and aspect prediction, optimization only toward relevance matching during fine-tuning undermines its ability to predict aspect values. In contrast, MADRAL and MURAL retain most of such ability after fine-tuning since they use extra aspect embeddings to perform aspect learning.

Lastly, for phrase-level evaluation, MURAL has the best aspect prediction accuracy. As we know, MURAL also has the best retrieval performance, which means MURAL can learn the two objectives well and let better aspect embeddings assist relevance matching more. Notably, the accuracy at the word and token level is not comparable with the phrase level since the ground truth is different. The finer-level prediction accuracy is also good. When the granularity becomes finer, the accuracy becomes lower after fine-tuning, which is probably because finer grains have more ground-truth values, making the multi-label classification more challenging.

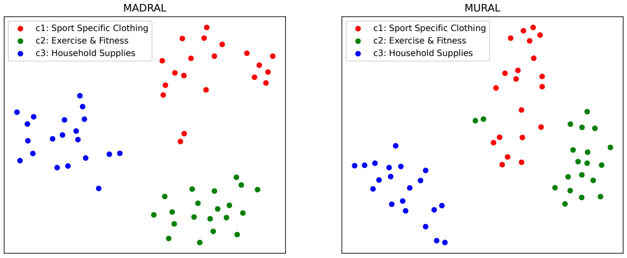


Figure 4: The t-SNE plot of item representations for MADRAL and MURAL on MA-Amazon.

6.5 Case Visualization

We visualize the item representations of three categories, as shown in Figure 1, to see their distributions in semantic space. Specifically, *c1* (Sport Specific Clothing) and *c2* (Exercise & Fitness) are semantically similar, while *c3* (Household Supplies) is unrelated to the first two. We first use MADRAL and MURAL to obtain all the item representations on MA-Amazon and put items into their categories. Then we randomly pick 20 items of *c1*, *c2* and *c3* and plot them using the t-SNE toolkit in Figure 4. We can observe that MADRAL separates *c1*, *c2*, and *c3* to a similar extent. By contrast, MURAL places the related categories *c1* and *c2* closer and puts them farther from the unrelated *c3*. This demonstrates MADRAL's inability to discern the semantic similarity between *c1* and *c2*, as it treats different phrase-level product categories as isolated IDs, overlooking their word-level semantic connections. MURAL can capture fine-grained semantic relations among similar aspect values while maintaining precise phrase-level aspect discrimination by incorporating both coarse and fine granularity information.

7 CONCLUSIONS

In this paper, we propose a multi-granularity-aware aspect learning model that enhances the utilization of additional aspect information in structured data. Unlike previous methods that disregard the semantic relationship among different aspect values, our approach incorporates multiple granularities of aspect values to facilitate query/item representation learning. By effectively capturing the semantics of queries/items from implicit views, our model achieves compelling performance even without the supervision of aspect annotations. Empirical results on two real-world datasets demonstrate the superiority of MURAL.

ACKNOWLEDGMENTS

This work was funded by the National Natural Science Foundation of China (NSFC) under Grants No. 62302486, the Innovation Project of ICT CAS under Grants No. E361140, the CAS Special Research Assistant Funding Project, the Lenovo-CAS Joint Lab Youth Scientist Project, and the project under Grants No. JCKY2022130C039. This work was also supported by Ant Group through Ant Innovative Research Program.

REFERENCES

[1] 2023. https://github.com/fxsjy/jieba.

[2] Qingyao Ai, Vahid Azizi, Xu Chen, and Yongfeng Zhang. 2018. Learning Heterogeneous Knowledge Base Embeddings for Explainable Recommendation. *Algorithms* 11, 9 (2018), 137.

[3] Qingyao Ai, Yongfeng Zhang, Keping Bi, and W. Bruce Croft. 2020. Explainable Product Search with a Dynamic Relation Embedding Model. *ACM Trans. Inf. Syst.* 38, 1 (2020), 4:1–4:29.

[4] Saeid Balaneshinkordan, Alexander Kotov, and Fedor Nikolaev. 2018. Attentive Neural Architecture for Ad-hoc Structured Document Retrieval. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018*, Alfredo Cuzzocrea, James Allan, Norman W. Paton, Divesh Srivastava, Rakesh Agrawal, Andrei Z. Broder, Mohammed J. Zaki, K. Selçuk Candan, Alexandros Labrinidis, Assaf Schuster, and Haixun Wang (Eds.). ACM, 1173–1182.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805

[6] Luyu Gao and Jamie Callan. 2021. Condenser: a Pre-training Architecture for Dense Retrieval. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (Eds.). Association for Computational Linguistics, 981–993.

[7] Luyu Gao and Jamie Callan. 2022. Unsupervised Corpus Aware Language Model Pre-training for Dense Passage Retrieval. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (Eds.). Association for Computational Linguistics, 2843–2853.

[8] Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2022. Tevatron: An Efficient and Flexible Toolkit for Dense Retrieval. *CoRR* abs/2203.05765 (2022). arXiv:2203.05765

[9] Jiafeng Guo, Yinqiong Cai, Yixing Fan, Fei Sun, Ruqing Zhang, and Xueqi Cheng. 2022. Semantic Models for the First-Stage Retrieval: A Comprehensive Review. *ACM Trans. Inf. Syst.* 40, 4 (2022), 66:1–66:42.

[10] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. *CoRR* abs/2004.04906 (2020). arXiv:2004.04906

[11] Omar Khattab and Matei Zaharia. 2020. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, Jimmy X. Huang, Yi Chang, Xueqi Cheng, Jaap Kamps, Vanessa Murdock, Ji-Rong Wen, and Yiqun Liu (Eds.). ACM, 39–48.

[12] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised Contrastive Learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (Eds.).

[13] Weize Kong, Swaraj Khadanga, Cheng Li, Shaleen Kumar Gupta, Mingyang Zhang, Wensong Xu, and Michael Bendersky. 2022. Multi-Aspect Dense Retrieval. In *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*, Aidong Zhang and Huzefa Rangwala (Eds.). ACM, 3178–3186.

[14] Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, Anna Korhonen, David R. Traum, and Lluís Màrquez (Eds.). Association for Computational Linguistics, 6086–6096.

[15] Jimmy Lin, Rodrigo Nogueira, and Andrew Yates. 2021. *Pretrained Transformers for Text Ranking: BERT and Beyond.* Morgan & Claypool Publishers.

[16] Jimmy Lin, Rodrigo Frassetto Nogueira, and Andrew Yates. 2021. *Pretrained Transformers for Text Ranking: BERT and Beyond.* Morgan & Claypool Publishers.

[17] Binsheng Liu, Xiaolu Lu, Oren Kurland, and J. Shane Culpepper. 2018. Improving Search Effectiveness with Field-based Relevance Modeling. In *Proceedings of the 23rd Australasian Document Computing Symposium, ADCS 2018, Dunedin, New Zealand, December 11-12, 2018.* ACM, 11:1–11:4.

[18] Shuqi Lu, Chenyan Xiong, Di He, Guolin Ke, Waleed Malik, Zhicheng Dou, Paul Bennett, Tie-Yan Liu, and Arnold Overwijk. 2021. Less is More: Pre-training a Strong Siamese Encoder Using a Weak Decoder. *CoRR* abs/2102.09206 (2021). arXiv:2102.09206

[19] Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. 2021. Sparse, Dense, and Attentional Representations for Text Retrieval. *Trans. Assoc. Comput. Linguistics* 9 (2021), 329–345.

[20] Xinyu Ma, Jiafeng Guo, Ruqing Zhang, Yixing Fan, and Xueqi Cheng. 2022. Pre-train a Discriminative Text Encoder for Dense Retrieval via Contrastive Span Prediction. In *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, Enrique Amigó, Pablo Castells, Julio Gonzalo, Ben Carterette, J. Shane Culpepper, and Gabriella Kazai (Eds.). ACM, 848–858.

[21] Xinyu Ma, Jiafeng Guo, Ruqing Zhang, Yixing Fan, Xiang Ji, and Xueqi Cheng. 2021. PROP: Pre-training with Representative Words Prediction for Ad-hoc Retrieval. *Proceedings of the 14th ACM International Conference on Web Search and Data Mining* (2021).

[22] Rutu Mulkar-Mehta, Jerry R. Hobbs, and Eduard H. Hovy. 2011. Granularity in Natural Language Discourse. In *Proceedings of the Ninth International Conference on Computational Semantics, IWCS 2011, January 12-14, 2011, Oxford, UK*, Johan Bos and Stephen Pulman (Eds.). The Association for Computer Linguistics.

[23] Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2021. RocketQA: An Optimized Training Approach to Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tür, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (Eds.). Association for Computational Linguistics, 5835–5847.

[24] Chandan K. Reddy, Lluís Màrquez, Fran Valero, Nikhil Rao, Hugo Zaragoza, Sambaran Bandyopadhyay, Arnab Biswas, Anlu Xing, and Karthik Subbian. 2022. Shopping Queries Dataset: A Large-Scale ESCI Benchmark for Improving Product Search. *CoRR* abs/2206.06588 (2022). arXiv:2206.06588

[25] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (Eds.). Association for Computational Linguistics, 3980–3990.

[26] Hongyu Shan, Qishen Zhang, Zhongyi Liu, Guannan Zhang, and Chenliang Li. 2023. Beyond Two-Tower: Attribute Guided Representation Learning for Candidate Retrieval. In *Proceedings of the ACM Web Conference 2023.* 3173–3181.

[27] Xiaojie Sun, Keping Bi, Jiafeng Guo, Xinyu Ma, Yixing Fan, Hongyu Shan, Qishen Zhang, and Zhongyi Liu. 2023. Pre-Training with Aspect-Content Text Mutual Prediction for Multi-Aspect Dense Retrieval. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management* (Birmingham, United Kingdom) *(CIKM '23)*. Association for Computing Machinery, New York, NY, USA, 4300–4304. https://doi.org/10.1145/3583780.3615157

[28] Amir Vakili Tahami, Kamyar Ghajar, and Azadeh Shakery. 2020. Distilling Knowledge for Fast Retrieval-based Chat-bots. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, Jimmy X. Huang, Yi Chang, Xueqi Cheng, Jaap Kamps, Vanessa Murdock, Ji-Rong Wen, and Yiqun Liu (Eds.). ACM, 2081–2084.

[29] Wilson L. Taylor. 1953. "Cloze Procedure": A New Tool for Measuring Readability. *Journalism & Mass Communication Quarterly* 30 (1953), 415 – 433.

[30] Shitao Xiao, Zheng Liu, Yingxia Shao, and Zhao Cao. 2022. RetroMAE: Pre-Training Retrieval-oriented Language Models Via Masked Auto-Encoder. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (Eds.). Association for Computational Linguistics, 538–548.

[31] Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. 2021. Approximate Nearest Neighbor Negative Contrastive Learning for Dense Text Retrieval. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021.* OpenReview.net.

[32] Hamed Zamani, Bhaskar Mitra, Xia Song, Nick Craswell, and Saurabh Tiwary. 2018. Neural Ranking Models with Multiple Document Fields. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM 2018, Marina Del Rey, CA, USA, February 5-9, 2018*, Yi Chang, Chengxiang Zhai, Yan Liu, and Yoelle Maarek (Eds.). ACM, 700–708.

[33] Shunyu Zhang, Yaobo Liang, Ming Gong, Daxin Jiang, and Nan Duan. 2022. Multi-View Document Representation Learning for Open-Domain Dense Retrieval. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (Eds.). Association for Computational Linguistics, 5990–6000.