REGENT: Relevance-Guided Attention for Entity-Aware Multi-Vector Neural Re-Ranking

Shubham Chatterjee
Missouri University of Science and Technology
Department of Computer Science
Rolla, Missouri, United States
shubham.chatterjee@mst.edu

ABSTRACT

Current neural re-rankers often struggle with complex information needs and long, content-rich documents. The fundamental issue is not computational–it is *intelligent content selection*: identifying what matters in lengthy, multi-faceted texts. While humans naturally anchor their understanding around key entities and concepts, neural models process text within rigid token windows, treating all interactions as equally important and missing critical semantic signals. We introduce REGENT, a neural re-ranking model that mimics human-like understanding by using entities as a "semantic skeleton" to guide attention. REGENT integrates relevance guidance directly into the attention mechanism, combining fine-grained lexical matching with high-level semantic reasoning. This relevance-guided attention enables the model to focus on conceptually important content while maintaining sensitivity to precise term matches. REGENT achieves new state-of-the-art performance in three challenging datasets, providing up to 108% improvement over BM25 and consistently outperforming strong baselines including ColBERT and RankVicuna. To our knowledge, this is the first work to successfully integrate entity semantics directly into neural attention, establishing a new paradigm for entity-aware information retrieval.

CCS CONCEPTS

• **Information systems** → **Information retrieval**; **Document representation**; **Retrieval models and ranking**;

KEYWORDS

Multi-Vector Reranking; Query-Specific Embedding; Relevance-guided Attention

ACM Reference Format:
Shubham Chatterjee. 2025. REGENT: Relevance-Guided Attention for Entity-Aware Multi-Vector Neural Re-Ranking. In *Proceedings of the 2025 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region (SIGIR-AP 2025), December 7–10, 2025, Xi'an, China.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3767695.3769476

SIGIR-AP 2025, December 7–10, 2025, Xi'an, China

ACM ISBN 979-8-4007-2218-9/2025/12
https://doi.org/10.1145/3767695.3769476

1 INTRODUCTION

Neural retrieval models perform well on benchmarks with short, factual queries and brief passages [31, 32, 63], achieving substantial gains over traditional methods on datasets such as MS MARCO [4] and the TREC Deep Learning tracks [7]. However, real-world search often involves complex queries that require multi-hop reasoning, synthesis of evidence, or understanding relationships across long, information-rich documents such as news articles or scientific papers. For example, answering the query "*Was the crash that followed the dot-com bubble an overreaction considering the ultimate success of the Internet?*" demands not just factual recall but also historical and interpretive reasoning. A human searching for an answer in a long article would naturally focus on themes like economic impact and the rise of internet businesses, while ignoring unrelated technical details. In contrast, neural models typically process only a limited portion of text (e.g., first 512 tokens in BERT) or divide documents into chunks that are scored independently before aggregation [36]. Each of these approaches risks missing globally relevant context, which can lead to degraded performance on complex queries.

We posit that the core challenge is not simply handling longer input, but performing *intelligent content selection*: identifying and focusing on the most relevant portions of complex documents. We argue that entities can play a central role in addressing this gap. Entities can be viewed as a high-level *semantic skeleton* that exposes the conceptual structure of a document. Entities can segment text into regions related to key topics or actors, providing a coarse but meaningful map of the content. For example, in a discussion of the dot-com bubble, entities like "`Nasdaq`," "`Amazon`," and "`venture capital`" highlight market dynamics and investment trends. Crucially, entity representations are computed independently of token windows, enabling models to capture salient context beyond local spans. Existing entity-aware methods [41, 66] typically rely on static, context-agnostic entity similarity matrices, treating all entities as equally important regardless of query. This does not capture the dynamic semantic alignment between the query and document entities that is essential for modeling complex queries. For instance, "`Amazon`" might be highly relevant for queries about e-commerce success stories, but less important for queries focused on the broader economic policy implications of the dot-com era.

In parallel, lexical signals (e.g., BM25), remain a strong source of fine-grained evidence, offering cues based on exact term overlap. Previous hybrid approaches have attempted to combine lexical and neural signals [3, 61], but typically do so post hoc, merging independently computed document level scores. This late stage fusion treats lexical and semantic signals as separate streams, preventing lexical cues from influencing the neural model's internal

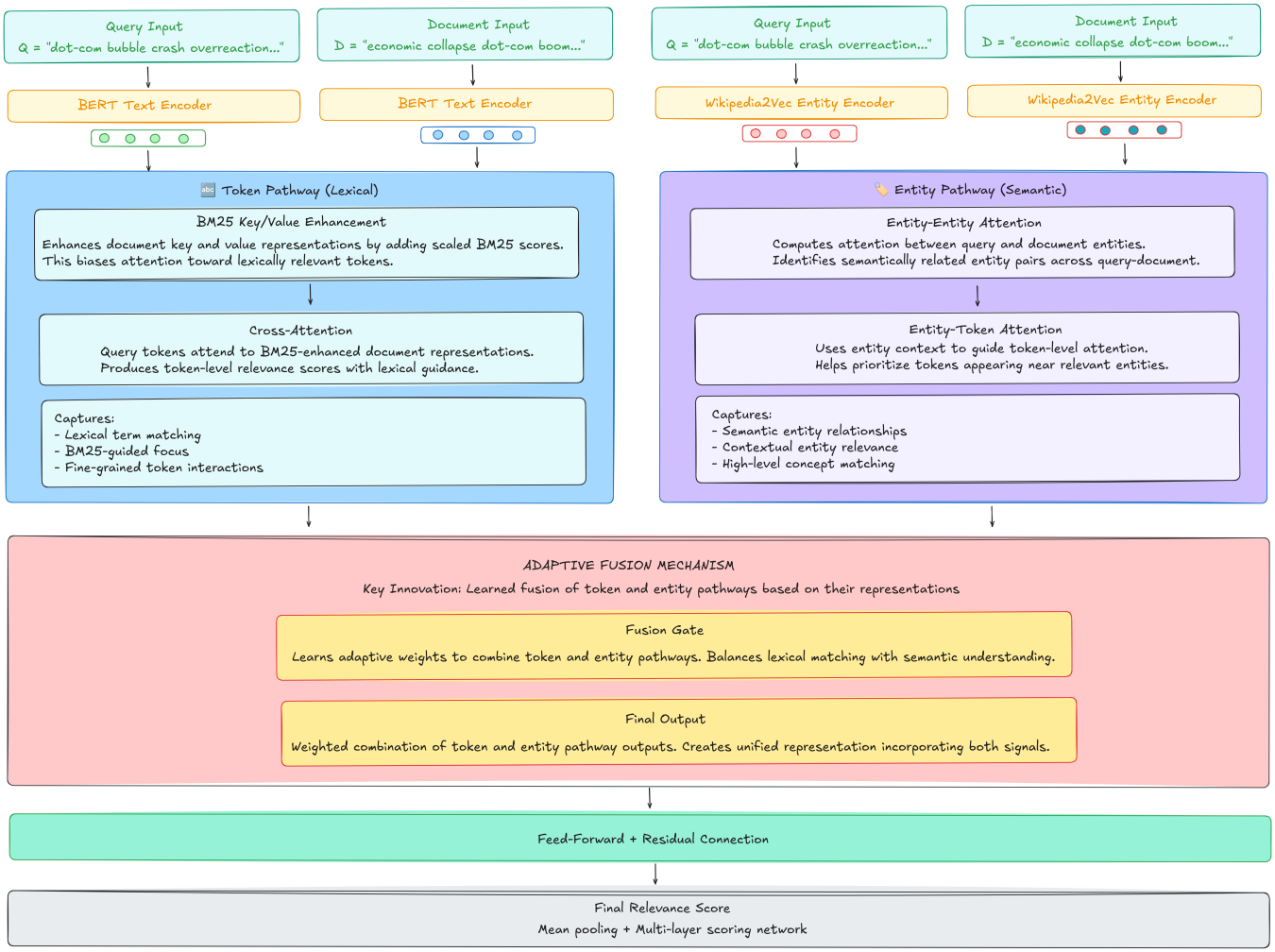


Figure 1: REGENT Architecture Overview. The model processes query and document inputs through separate BERT encoders to generate contextual embeddings. REGENT employs a dual-pathway attention mechanism: (1) The token pathway enhances document key and value representations with BM25 scores, then applies cross-attention to capture lexical matches between query and document tokens. (2) The entity pathway first projects pre-computed entity embeddings to the hidden dimension, computes entity-entity attention to identify semantically related concepts, then uses this entity context to guide token-level attention. An adaptive fusion mechanism learns to combine both pathways, balancing lexical matching with semantic understanding. The final output undergoes feed-forward processing with residual connections before mean pooling and scoring to produce a relevance score. This architecture enables fine-grained integration of traditional IR signals (BM25) with neural semantic reasoning through entities, moving beyond post-hoc score combination to embedded relevance guidance within the attention mechanism itself.

representation learning. We argue that lexical signals such as BM25 should not simply be combined with neural output, but should *actively guide the neural ranking process itself*. This requires moving beyond document-level fusion and *integrating relevance signals directly into the attention mechanism*, where neural models form their understanding of query-document relationships. Since attention operates at the token level, such an integration necessitates token-level BM25 guidance. This insight leads to our core contribution: **relevance-guided attention**—a mechanism that biases attention computations using explicit relevance signals.

We instantiate this idea in REGENT (REelevance-Guided ENTity-aware reranker).[1] REGENT is built on a multi-vector architecture designed to process two distinct types of information in parallel: one pathway handles the fine-grained vectors representing document tokens, while a second pathway processes the high-level vectors representing semantic entities. It then guides attention using two

[1] **Code and data available at:https://github.com/shubham526/SIGIR-AP-2025-REGENT**

complementary signals: token-level BM25 scores to highlight lexically relevant terms and query-specific entity representations to focus on semantically important concepts.

A multi-vector architecture naturally supports this dual guidance, preserving fine-grained token interactions while enabling coordinated entity-level processing. When processing the dot-com query, REGENT learns to prioritize passages mentioning relevant entities like "Nasdaq" and "venture capital", while the BM25 signals simultaneously highlight exact matches such as bubble and crash. These lexical and semantic signals work together to direct attention toward the most relevant content, even when it is scattered across a long document. We show that REGENT achieves significant improvements across three large-scale datasets featuring long-form news and social science articles paired with complex information needs. The model outperforms strong neural baselines including recent LLM-based re-rankers such as RankVicuna [51] and RankZephyr [52] with gains of up to 108% over BM25. Crucially, removing the model's entity component–which captures relationships between key entities–results in a 74% drop in performance, underscoring the essential role of the semantic skeleton in effective long-document retrieval.

Contributions. We make three key contributions to neural IR: (1) **Relevance-guided attention:** the first mechanism to directly weave lexical and entity signals into attention computations, shifting from post-hoc score combination to integrated neural guidance during ranking; (2) **Token-level BM25 integration:** a principled method for incorporating fine-grained lexical evidence into transformer attention, enabling models to focus on individually relevant terms rather than coarse document-level signals; and (3) **Dynamic entity-aware processing:** a dual-pathway architecture that learns query-specific entity relationships through dedicated attention layers, moving beyond static similarity matrices to context-sensitive semantic reasoning.

The remainder of this paper is structured as follows. We first review related work in entity-oriented search and neural IR in Section 2. We then detail the REGENT architecture, including its relevance-guided attention mechanism, in Section3. Our experimental setup, datasets, and baselines are described in Section4. We present and analyze our results in Section 5, including a series of ablation studies and qualitative analyses. Finally, we conclude our work in Section 6.

2 RELATED WORK

Entity-Oriented Search. Early work such as EQFE [12] leveraged entity-based features, demonstrating the utility of structured signals. This evolved into latent semantic projection approaches [19, 39], where queries and documents were mapped to an entity space to capture hidden semantic relationships. The move toward explicit entity-term integration began with early entity-based language models [17, 53], which balanced term- and entity-level signals. The Word-Entity Duet [65] introduced rich four-way interactions between queries and documents, while Explicit Semantic Ranking [67] employed knowledge graphs for structured matching. Approaches such as EDRM [41] incorporate knowledge graph semantics into neural ranking models through pre-computed entity similarities.

Neural Information Retrieval. The evolution of neural IR can be traced through several paradigmatic shifts. Pre-BERT approaches followed two main trajectories: representation-based models that used static embeddings [24, 44, 56] and interaction-based models that computed term-level similarity matrices [22, 25, 66]. The introduction of BERT [13] marked a transformative moment, spawning innovations like cross-encoders for fine-grained query-document interaction [1, 9] and bi-encoders such as DPR [31] and ANCE [68] that balanced effectiveness with efficiency.

Recent advances have moved beyond single-vector representations to capture richer query-document interactions. In this regard, ColBERT [32] pioneered "late interaction" mechanisms using dense token-level representations while poly-encoders [26] generate multiple document representations through learned context codes.

Neural approaches have also transformed pseudo-relevance feedback by leveraging embedding-based similarity measures. While ANCE-PRF [70] enhances retrieval effectiveness by retraining the query encoder using PRF information, ColBERT-PRF [62] directly utilizes BERT embeddings for retrieval without further training, avoiding topic drift issues common with polysemous words.

Hybrid Retrieval. Prior work has also shown how to blend traditional (sparse) IR signals with transformers. DeepCT [10], docT5query [49], and CEQE [45] identify important terms or expansions using contextualized models. While the original DPR paper [31] reported limited gains from combining dense and sparse signals, subsequent work has shown that hybrid approaches can consistently outperform purely dense or sparse models. Ma et al. [42] demonstrated that carefully designed dense-sparse hybrids yield significant improvements across multiple benchmarks. CLEAR [20] proposed a jointly trained hybrid where the dense encoder complements BM25 by capturing semantic matches it misses. TCT-ColBERT [38] used knowledge distillation with ColBERT as a teacher to produce soft labels on the fly, combining these with BM25 and doc2query-T5 expansion for strong effectiveness with high efficiency. Askari et al. [3] showed that injecting BM25 scores as special tokens into transformer inputs improves re-ranking, offering direct evidence that lexical signals can be leveraged within neural models. Recent advancements include SPLADE-v3 [34], which improves lexical expansion with better regularization, and LexMAE [55], which introduces pre-training objectives that preserve lexical sensitivity. Meanwhile, hybrid systems have begun incorporating LLMs, as seen in HyDE [21] and Promptagator [11], which use hypothesized documents or synthetic queries to improve retrieval.

LLM-based Re-Rankers. LLMs have recently shown strong zero-shot reranking performance. RankVicuna [51] achieved near-GPT-3.5 results on TREC DL using a 7B model. RankZephyr [52] further closed the gap with GPT-4 while demonstrating robust generalization across BEIR and NovelEval. More advanced methods such as RankR1 [74] introduced reasoning via reinforcement learning, yielding large gains on complex and out-of-domain queries with minimal training data. Search-R1 [28] extended this by teaching LLMs to iteratively generate and refine search queries during reasoning, improving performance on retrieval-augmented QA tasks.

Advances in Training and Representation. RetroMAE [63] introduced retrieval-specific masked autoencoding, outperforming general language modeling by aligning training objectives with retrieval tasks. E5 [60] extended this with weakly supervised

contrastive learning, achieving strong zero-shot performance and demonstrating the effectiveness of large-scale contrastive pre-training for general-purpose embeddings. Instruction-based models such as INSTRUCTOR [57] added architectural flexibility by generating embeddings conditioned on natural language prompts, thus enabling a single model to adapt across retrieval scenarios without task-specific fine-tuning.

3 APPROACH

Given a complex query Q and a set of long candidate documents $\mathcal{D}$, our goal is to *re-rank* the candidates by their relevance to Q. We propose REGENT, a re-ranking model built around a novel **relevance-guided attention mechanism**. This is our key innovation. Unlike standard attention, which relies only on learned token interactions, our approach explicitly guides attention using two relevance signals: (1) **token-level BM25 scores** that enhance key and value representations for lexically matched terms, and (2) **query-specific entity representations** that modulate attention weights based on semantic relevance. This allows the model to integrate lexical and semantic signals directly during attention computation, rather than fusing them post hoc, as done by prior work [3, 61]

3.1 Model Architecture

At its core, REGENT uses a pre-trained `bert-base-uncased` encoder [13] to encode queries and documents separately. Inputs are padded or truncated to 512 tokens. These embeddings are passed through two cross-attention layers (each with eight heads), where queries attend to document tokens *via an entity-aware attention module*, followed by a feed-forward network. This module integrates BM25 scores and entity signals to guide attention. The final query representation is mean-pooled and fed into a multi-layer scoring network with residual connections, `GELU` activations, and `LayerNorm`, producing a single relevance score via a linear output layer.

3.2 Token-Level BM25 Integration

BM25 Score Alignment with Subword Tokens. Given a query-document pair, we compute a ranking signal vector $r \in \mathbb{R}^n$ where each element r_i represents the BM25 score for the i-th token position. These BM25 scores are computed using Lucene's EnglishAnalyzer, which applies standard IR preprocessing including stopword removal, stemming, and lowercasing. To bridge the gap between this preprocessing and BERT's tokenization, we track how each word splits into subword tokens. During indexing, we store a tuple (w_i, s_i, e_i) where w_i is the word index, and s_i, e_i denote the start and end positions of its subword tokens. To obtain token-level BM25 scores, we first compute term-level scores and then propagate them to all corresponding subword tokens. For example, both "play" and "##ing" inherit the BM25 score of "playing".

Why Not Tokenize BM25 Terms Directly? We considered an alternative approach of re-tokenizing Lucene's preprocessed (e.g., stemmed) terms with the BERT tokenizer and assigning BM25 scores directly to the resulting subword units. However, this strategy is conceptually problematic because BM25 is defined over full words or terms, not arbitrary subword pieces like "##ing" or "##ly," which lack independent frequency statistics or meaningful retrieval signals. Therefore, propagating word-level scores across subword spans, as done in our approach, is a more principled and interpretable integration of BM25 into subword-based neural architectures. This approach ensures that stopwords and punctuation retained by BERT (e.g., "a", "the", "of") receive minimal or zero BM25 scores since they were removed or heavily downweighted in the Lucene preprocessing pipeline. Consequently, these tokens have negligible impact on the attention mechanism despite being present in BERT's representation.

Integrating BM25 into Attention via Key and Value Enhancement. We use these token-level scores to improve key representations (K) and value (V) representations in the attention mechanism. In a standard attention mechanism, Key (K) and Value (V) matrices are learned projections of the input token embeddings. Our approach enhances these matrices to make lexically important tokens stand out more, influencing both which tokens are attended to (via keys) and how much they contribute to the output (via values). Specifically, let $R \in \mathbb{R}^{n \times d}$ be the matrix formed by repeating the BM25 score vector r for each dimension of the embedding space, where d is the embedding dimension. We then compute $K' = K + \alpha \cdot R$ and $V' = V + \alpha \cdot R$. Here, $K, K' \in \mathbb{R}^{n \times d}$ are the original and enhanced key matrices, $V, V' \in \mathbb{R}^{n \times d}$ are the original and enhanced value matrices, and $\alpha \in \mathbb{R}$ is a learnable scalar parameter that controls the influence of BM25 scores on the attention mechanism. Our intuition is that this enhancement would bias key representations toward strong lexical matches and enriches value representations to emphasize high-BM25 tokens, while minimizing the influence of less informative tokens. By influencing both attention weights (via keys) and contextual representations (via values), the model would be able to learn matching patterns that effectively integrate traditional lexical signals with neural semantic evidence.

3.3 Query-Specific Entity Set Construction

As discussed in Section 1, entities can provide a meaningful semantic scaffold to organize document content. However, using all entities indiscriminately can introduce noise and dilute the focus of the attention mechanism. To address this, we design a multi-stage pipeline that builds a focused *query-specific* set of entity representations. First, we retrieve the top 1,000 documents using BM25 and aggregate all unique entities linked within these documents to form a candidate pool. Next, we train a separate BERT-based entity ranking model (via 5-fold CV) to score each candidate entity's relevance to the query. This ranker is a standard BERT cross-encoder. For each query-entity pair, we format the input as "`[CLS]` query text `[SEP]` entity name `[SEP]`". The final relevance score is derived from the `[CLS]` token's output embedding, which is passed through a linear layer. This ranker is trained once and reused at inference time to efficiently score entities given a new query. Since the candidate pool remains fixed, we can precompute document-entity mappings and quickly compute query-document relevance scores using only the top-ranked entities.

To train the ranker, we follow established practice [14, 15, 47], treating entities found in documents marked relevant in the qrels as positive examples and all others as negative. Using the trained ranker, we then score all candidate entities for each query and

retain the top 20 as the query-relevant entity set. For each query-document pair, we take the intersection of the document's entities with this set and scale each entity's pre-trained Wikipedia2Vec [69] embedding by its corresponding ranker score. This results in a *query-specific* set of entity embeddings that REGENT's entity-aware attention mechanism (described below) can use to focus on semantically meaningful signals.

3.4 Entity-Aware Dual-Pathway Attention

We employ parallel token and entity attention pathways to enhance relevance modeling. The token pathway captures lexical matches using BM25 signals:

$$A_t = \text{softmax}\left(\frac{QK'^T}{\sqrt{d_k}}\right)V' \tag{1}$$

where Q represents query token embeddings, K', V' are the BM25-enhanced document representations, and d_k is the scaling factor.

The entity pathway processes the refined entity sets generated by the pipeline described in Section 3.3. It establishes semantic context in two steps: First, it computes attention between the query's entity set and the document's (filtered) entity set to identify relevant entity matches:

$$A_e = \text{softmax}\left(\frac{E_q W_q^e (E_d W_k^e)^T}{\sqrt{d_k}}\right)E_d W_v^e \tag{2}$$

where $E_q \in \mathbb{R}^{n_q \times d_e}$ and $E_d \in \mathbb{R}^{n_d \times d_e}$ are query and document entity embeddings, respectively, which have been prescaled by their relevance scores as determined by our entity ranking pipeline. W_q^e, W_k^e, W_v^e are learnable query, key, and value projections specific to entity-entity attention. We implement this using multi-head attention to capture different types of entity relationships.

Then, it allows these entity matches to influence token representations through a separate attention mechanism:

$$A_{et} = \text{softmax}\left(\frac{QW_q^t (A_e W_k^t)^T}{\sqrt{d_k}}\right)A_e W_v^t \tag{3}$$

where Q represents query token embeddings, and W_q^t, W_k^t, W_v^t are learnable projections specifically for entity-token interactions. Importantly, the projection matrices used in equations (2) and (3) are *not* shared ($W_q^e \neq W_q^t$, $W_k^e \neq W_k^t$, $W_v^e \neq W_v^t$) because they serve fundamentally different purposes: the former transforms entity embeddings for entity-entity matching, while the latter projects entity attention outputs for token-level contextualization. This allows the model to learn distinct transformation patterns for each attention mechanism: one optimized for entity relationship detection and another for incorporating entity context into token representations.

Our intuition is that this two-step process enables context-aware token matching. First, A_e aligns related entities across the query and document (e.g., linking "`Nasdaq`" in the query to "`tech stocks`" in the document). Then, A_{et} uses this entity context to guide token attention —- helping the model prioritize mentions of "crash" or "valuation" that appear near relevant entities like "`Amazon`" or "`venture capital`", rather than in unrelated sections.

3.5 Adaptive Fusion Mechanism

The token and entity pathways are combined through a fusion mechanism. First, we compute attention weights:

$$\alpha = \sigma(\text{LayerNorm}(W_f[A_t; A_{et}])) \tag{4}$$

where σ is the sigmoid function, W_f is a learnable projection, and $[;]$ denotes concatenation. The final output is then computed as:

$$O = \alpha \odot A_t + (1 - \alpha) \odot A_{et} \tag{5}$$

Although both tokens and entities can have BM25 scores, we integrate them through separate pathways to reflect their distinct roles: tokens capture lexical matches, while entities represent higher-level semantic concepts. Token scores reflect term frequency, while entity relevance depends on contextual importance. Our dual-pathway design leverages this distinction—lexical cues guide token attention, while entity context informs semantic focus. An adaptive fusion layer balances the two; our experiments show that this separation outperforms unified models.

4 EXPERIMENTAL SETUP

4.1 Datasets

We evaluate REGENT on three benchmarks selected for their (1) long-form documents with high entity density and (2) complex, non-factual information needs requiring semantic reasoning. **CODEC** [43] features 42 social science queries over 729K entity-linked documents (avg. 159 entities per document), with 6,186 expert-annotated relevance judgments. **TREC Robust 2004** [59] includes 250 challenging queries designed to stress lexical models, covering 528K articles (avg. 116 entities per document) and 311,409 graded relevance labels. We specifically use the "title" queries, which is a standard and challenging sub-task for this collection **TREC Core 2018** [2] contains 50 complex queries over 595K news/blog posts (avg. 123 entities per document) with 26,233 graded judgments.

We deliberately exclude benchmarks like MS MARCO and TREC DL, as their focus on short passages and factoid queries with minimal entity presence (e.g., MS MARCO averages 1.7–2.1 entities per document, and over 97% of its queries mention at most one entity [29]) makes them fundamentally ill-suited for evaluating the core contribution of REGENT, which is its ability to reason over rich entity relationships in long documents. Our selected datasets follow established practice in entity-oriented IR [12, 46] and better capture the challenges of long-document retrieval

Evaluation. We report Prec@20, nDCG@20, and Mean Average Precision (MAP) using the official `trec_eval` tool (using `-c` flag).

4.2 Baselines

To validate REGENT, we compare it against a broad set of state-of-the-art neural re-ranking models. All fine-tuned models are trained on the target datasets using the same 5-fold cross-validation as REGENT for fair comparison. Baselines span five main categories:

1. Cross-Encoder Models. These models represent the standard and powerful paradigm of fine-tuning a transformer for point-wise re-ranking by processing a concatenated `[CLS] query [SEP] document` sequence. We include a wide range of established and recent architectures: **BERT** [13], **RoBERTa** [40], **DeBERTa** [23], **ELECTRA** [6], **ConvBERT** [27], and the sequence-to-sequence

RankT5 [73]. We also include several zero-shot cross-encoders from the Sentence-BERT library [54].

2. Bi-Encoder and Multi-Vector Models. We include a fine-tuned **ColBERT v2** [32] model. We also test a suite of bi-encoder models, including a zero-shot version of **DPR** [31] and various pre-trained models from the Sentence-BERT library to represent the efficient dual-encoder paradigm.

3. Entity-Aware Models. To compare specifically with other methods that leverage entity knowledge, we include two key baselines. **EDRM** [41] is a pioneering neural model that uses pre-computed entity similarity matrices. **ERNIE** [71] is a knowledge-enhanced pre-trained model that integrates entity information directly during its pre-training phase. These models allow us to assess the benefits of REGENT's dynamic, relevance-guided entity processing against more static approaches.

4. LLM-Based Re-Rankers. To situate REGENT's performance against the latest LLM-based rerankers, we include several powerful, publicly available zero-shot re-rankers. This includes the pointwise **BGE-ReRanker-v2-Gemma** [5], **INSTRUCTOR** [57], and the listwise **RankVicuna** [51] and **RankZephyr** [52], which have demonstrated performance competitive with proprietary models.

5. Hybrid Models. Finally, to contrast REGENT's deep signal fusion with other hybrid approaches, we include several models that also combine lexical and semantic signals. We re-implement the work by Askari et al. [3] that injects BM25 scores directly into the input text sequence. We also re-implement work by Wang et al. [61] that linearly interpolates the final scores from a dense model and BM25. Lastly, we include several strong **Coordinate Ascent** baselines, which iteratively optimize the weights for combining BM25 with scores from various fine-tuned and zero-shot models.

Additional Baselines on TREC Robust 2004. Given the historical significance of the TREC Robust 2004 track, we include several additional influential full retrieval (1st stage, not reranking) models to provide a broader context for our results on this specific dataset. The results are shown in Table 2.

Note. We acknowledge that the field of LLM-based re-ranking is evolving rapidly; for instance, RankR1 [74] and Search-R1 [28] were published concurrently with our work. Our selection of baselines therefore focuses on a representative and diverse set of strong, publicly available models at the time of our experiments.

4.3 Implementation Details

Model Details. We implement our model in PyTorch using the HuggingFace library with `bert-base-uncased` as the base encoder (parameters not frozen). We fine-tune the model using binary cross-entropy loss with Adam [33] optimizer at a learning rate of 2e−5 and apply a linear warmup over the first 1,000 steps. Training is performed for 10 epochs with a batch size of 8, and early stopping based on validation MAP (computed via `pytrec_eval`). We apply gradient clipping and a dropout rate of 0.1 to prevent overfitting. Document embeddings and entity links are precomputed and cached for efficient inference. For all re-ranking experiments, the candidate set consists of the top 1000 documents retrieved by the initial BM25 ranking. This same set was provided to REGENT and all re-ranking baselines to ensure a fair comparison.

Train and Test Data. As positive examples during training, we use documents that are assessed as relevant in the ground truth provided with the dataset. Following the standard [31], for negative examples, we use documents from a BM25 candidate ranking (obtained using Pyserini default) which are either explicitly annotated as negative or not present in the ground truth. We balance the training data by keeping the number of negative examples the same as the number of positive examples. These examples are then divided into 5-folds for cross-validation. We create these folds at the query level.

Entity Linking and Embeddings. We use pre-trained Wikipedia2Vec embeddings [69] from Kamphuis et al. [30] as base entity representations and generate initial links using WAT [50], an open-source, deterministic EL system. We prefer WAT over LLM-based methods to ensure reproducibility, scalability, and avoid proprietary/API constraints. Recent work [16, 58, 64] also shows that traditional EL systems outperform LLMs, especially for rare or ambiguous entities that LLMs often misidentify. As described in Section 3.3, these initial links are refined through an entity selection and classification pipeline to produce a query-specific, relevance-weighted entity set for REGENT. Although we use standard tools, REGENT is EL-agnostic: its architecture does not rely on any specific linking method, allowing future EL improvements to be easily incorporated.

5 RESULTS AND ANALYSIS

As shown in Table 1, REGENT decisively establishes a new state-of-the-art for re-ranking on all three benchmarks. Performance gains are most pronounced on TREC Robust04, where REGENT achieves a MAP of 0.609, a remarkable 108% relative improvement over the strong baseline BM25, and an nDCG@20 of 0.785.

Crucially, these gains are not limited to a single class of models. REGENTT consistently surpasses a wide array of strong baselines, including fine-tuned cross-encoders like DeBERTa, multi-vector models like ColBERT v2, recent zero-shot LLM re-rankers such as RankZephyr, and other state-of-the-art hybrid models. This consistent and significant outperformance provides strong evidence that REGENT's fine-grained integration of lexical signals and entity context offers a more effective path to relevance modeling than what is possible with existing architectures.

To provide broader context for our results on the historically significant TREC Robust 2004 benchmark, we also compare REGENT's re-ranking performance against several influential full retrieval (first-stage) models in Table2. While re-rankers and full retrievers serve different functions in a multi-stage system, this comparison highlights the substantial effectiveness gains REGENT provides over the initial BM25 ranking it refines. It demonstrates that our re-ranking approach elevates performance far beyond what classic, standalone retrieval methods could achieve on this challenging task.

In the following sections, we analyze and discuss the results for the TREC Core 2018 dataset due to lack of space but similar results were obtained on all datasets.

5.1 Effect of Model Components

We first ask: **RQ1: What is the individual contribution and synergistic effect of the entity-based semantic pathway and**

Table 1: Results for re-ranking on Robust04 (Title), Core18, and CODEC datasets. Statistical significance is determined using paired t-tests ($p < 0.05$) with Bonferroni correction (▲ indicates significantly better, ▼ indicates significantly worse than BM25).

	TREC Robust04 (Title)			TREC Core 2018			CODEC		
	MAP	**nDCG@20**	**P@20**	**MAP**	**nDCG@20**	**P@20**	**MAP**	**nDCG@20**	**P@20**
BM25	0.292	0.435	0.384	0.315	0.447	0.459	0.363	0.380	0.432
Cross-Encoder Models (Fine-Tuned)									
RankT5 [73]	0.303	0.494▲	0.429▲	0.216▼	0.326▼	0.334▼	0.380	0.437	0.452▲
MonoBERT [48]	0.297	0.479	0.409	0.302	0.469	0.461	0.382	0.440	0.463▲
RoBERTa [40]	0.290	0.474	0.410	0.260▼	0.361▼	0.388▼	0.359▼	0.377▼	0.433▲
DeBERTa [23]	0.293	0.486	0.422	0.346	0.519▲	0.507	0.390	0.449	0.463▲
ELECTRA [6]	0.268	0.446	0.387	0.240▼	0.353	0.361	0.323▼	0.317▼	0.385▼
ConvBERT [27]	0.321▲	0.519▲	0.451▲	0.321	0.496	0.489	0.382	0.441	0.462▲
ERNIE [71]	0.289	0.475	0.412	0.340	0.520	0.507	0.391	**0.457**	0.474▲
EDRM [41]	0.067▼	0.102▼	0.094▼	0.092▼	0.098▼	0.134▼	0.298▼	0.289▼	0.352▼
SentenceBERT Cross-Encoders (Zero-Shot)									
ms-marco-MiniLM-L6-v2 [54]	0.275	0.447	0.392	0.272	0.434	0.429	0.365	0.426	0.436
ms-marco-electra-base [54]	0.233▼	0.405	0.346	0.175▼	0.289▼	0.306▼	0.367	0.423	0.447
qnli-distilroberta-base [54]	0.067▼	0.105▼	0.103▼	0.047▼	0.036▼	0.046▼	0.298▼	0.294▼	0.347▼
quora-distilroberta-base [54]	0.043▼	0.035▼	0.037▼	0.061▼	0.055▼	0.077▼	0.257▼	0.195▼	0.269▼
stsb-distilroberta-base [54]	0.045▼	0.037▼	0.038▼	0.061▼	0.057▼	0.076▼	0.260▼	0.195▼	0.276▼
Bi-Encoder Models									
ColBERT v2 [32] (Fine-Tuned)	0.292	0.473▲	0.410▲	0.267▼	0.455▲	0.451▼	0.375▲	0.446▲	0.456▲
DPR [31] (Zero-Shot)	0.170▼	0.300▼	0.259▼	0.210▼	0.309▼	0.322▼	0.338▲	0.364	0.401▲
SentenceBERT Bi-Encoders (Zero-Shot)									
all-MiniLM-L6-v2 [54]	0.222▼	0.394	0.334	0.264	0.419	0.419	0.365	0.412	0.420
multi-qa-MiniLM-L6-cos-v1 [54]	0.223▼	0.396	0.333	0.257	0.405	0.409	0.369	0.433	0.446
paraphrase-albert-small-v2 [54]	0.177▼	0.341▼	0.287▼	0.203▼	0.316▼	0.329▼	0.352▼	0.392	0.410▼
msmarco-bert-base-dot-v5 [54]	0.234▼	0.405	0.347	0.257▼	0.389	0.402	0.365	0.414	0.413
multi-qa-mpnet-base-dot-v1 [54]	0.340	0.500	0.449	0.314	0.451	0.465	0.263	0.205	0.278
LLM-Based (Zero-Shot)									
BAAI/bge-reranker-v2-gemma [5] (Pointwise)	0.266▼	0.504▲	0.431▲	0.322	0.479	0.479	0.263▼	0.205▼	0.278▼
INSTRUCTOR [57] (Pointwise)	0.266	0.459	0.393	0.301	0.482	0.481	0.396	0.468	0.478
RankVicuna [51] (Listwise)	0.296	0.467	0.400	0.315	0.446	0.459	0.382	0.426	0.456
RankZypher [52] (Listwise)	0.318▲	0.505▲	0.433▲	0.298	0.413	0.426	0.400▲	0.457▲	0.465
Hybrid Models (Trained)									
BM25InjectionReranker [3]	0.378▲	0.559▲	0.501▲	0.334	0.481	0.476	0.362▲	0.378	0.438▲
BM25DenseInterpolationReranker [61]	0.340▲	0.500▲	0.449▲	0.314	0.451	0.465	0.361	0.433	0.376
Coordinate Ascent (MonoBERT + BM25)	0.341▲	0.514▲	0.449▲	0.358▲	0.493▲	0.495▲	0.393▲	0.453▲	0.471▲
Coordinate Ascent (ELECTRA + BM25)	0.324▲	0.489▲	0.430▲	0.337▲	0.471	0.475	0.362	0.382	0.436
Coordinate Ascent (all-MiniLM-L6-v2 + BM25)	0.313▲	0.474▲	0.415▲	0.345▲	0.488	0.493	0.385▲	0.445▲	0.460▲
Coordinate Ascent (ms-marco-MiniLM-L6-v2 + BM25)	0.323▲	0.480▲	0.424▲	0.349▲	0.489	0.494	0.376	0.421▲	0.439▲
Coordinate Ascent (BAAI/bge-reranker-v2-gemma + BM25)	0.326▲	0.516▲	0.443▲	0.378▲	0.525▲	0.523▲	0.413▲	0.474▲	0.492▲
TREC Best	0.332▲	–	–	0.432▲	–	0.612▲	–	–	–
REGENT	**0.609▲**	**0.785▲**	**0.765▲**	**0.516▲**	**0.627▲**	**0.645▲**	**0.498▲**	0.423▲	**0.559▲**

token-level lexical guidance within REGENT's relevance-guided attention mechanism? To answer this, we conducted a series of ablation studies. The results (Table 3) confirm that both the entity-based semantic pathway and the token-level lexical guidance are critical, but they play distinct and complementary roles in the model's success.

The Primacy of the Semantic Skeleton. First, we assessed the individual contributions of the entity and BM25 signals. Removing the entity information pathway entirely caused a catastrophic performance drop, with MAP decreasing from 0.516 to 0.134 (–74%).[2] This is by far the most significant impact of any single component, validating our central hypothesis that the "semantic skeleton" provided by the entity context is not merely an auxiliary feature but the primary driver of REGENT's ability to perform deep semantic reasoning. Without it, the model loses its ability to understand the conceptual landscape of a document.

[2]While one might expect this variant to perform similarly to a standard cross-encoder, its architecture is not identical. It retains the cross-attention and fusion mechanism structures of REGENT, which are not optimized to function without the entity pathway, leading to sub-optimal performance compared to a purpose-built cross-encoder like MonoBERT.

Table 2: Results on TREC Robust 2004 title queries comparing REGENT to full retrieval methods. Note: REGENT is *reranking* the BM25 full retrieval results presented at the top.

Method	MAP	nDCG@20	P@20
BM25	0.29	0.44	0.38
BERT-QE [72]	0.39▲	0.55▲	0.49▲
CEQE [45]	0.31▲	0.46▲	0.40▲
DeepCT [10]	0.20▼	0.38▼	0.33▼
NPRF [35] [35]	0.29	0.45	0.41
EQFE [12]	0.33	0.43	0.38
ANCE-MaxP [68]	0.13▼	0.31▼	0.25▼
BERT-MaxP [9]	0.31	0.48	0.42
TCT-ColBERT [37]	0.23	0.43	0.36
PARADE [36]	0.30	0.52	0.45
SPLADE v2 [18]	0.22▼	0.42	0.36
REGENT	**0.61▲**	**0.79▲**	**0.77▲**

Table 3: Ablation study for architectural choices. Results reported on TREC Core 2018.

Model Variant	MAP	nDCG@20	P@20
RENET(No Entities)	0.134▼	0.174▼	0.189▼
REGENT(No BM25)	0.449▼	0.561▼	0.605▼
REGENT(Full)	**0.516**	**0.627**	**0.645**

In contrast, removing only the token-level BM25 signal also led to a notable performance decline (MAP from 0.516 to 0.449). Although smaller, this drop indicates that fine-grained lexical signals provide valuable complementary information that the entity pathway alone cannot capture. This suggests that the BM25 signal acts as a powerful "fine-tuner," helping the model ground its high-level semantic understanding in specific evidence-bearing term matches.

Granularity is Crucial: Token-Level vs. Document-Level BM25. To further validate our architectural choice of fine-grained signal integration, we investigated the impact of using a single document-level BM25 score instead of our token-level approach. In this variant, the model first computes a neural score using the entity-aware attention layers (with token-level BM25 disabled), then combines this score with the document's overall BM25 score using a learned linear transformation: score $= W[s_{\text{neural}}; s_{\text{BM25}}]$.

This experiment allows us to directly assess how the granularity of the BM25 signal affects performance. The results are striking: the document-level approach leads to a substantially lower MAP of 0.418. More revealingly, this is even worse than the model with no BM25 signal at all (MAP 0.449). This finding suggests that a single coarse-grained BM25 score, when naively aggregated, can be counterproductive, potentially confusing the model more than helping it. It provides strong evidence that for a lexical signal to be effective in a multi-vector attention framework, it must be integrated at the token level where it can directly guide fine-grained interactions.

Takeaway. Taken together, these ablations paint a clear picture. The entity pathway provides the foundational semantic understanding, while token-level BM25 integration provides precise lexical guidance. While both are important, the model's strength lies in processing them at the right granularity and fusing them effectively. Crucially, the sharp performance gap—74% drop without entities vs. 13% without BM25—shows that entity signals form a robust semantic backbone, retaining most of the model's performance on their own. This highlights that semantic understanding is essential for complex retrieval, and lexical matching alone is insufficient.

Table 4: Ablation study for fusion choices. Results reported on TREC Core 2018.

Method	MAP	nDCG@20	P@20
REGENT (Additive)	0.497▼	0.619	0.646
REGENT (Attention-based)	0.384▼	0.491	0.494
REGENT (Equal weighting)	0.496▼	0.618	0.644
REGENT (Gated GELU)	0.502	0.628	0.648
REGENT (Hard switch)	0.426▼	0.565▼	0.600
REGENT (Learned tanh)	0.498	0.617	0.655
REGENT (Learned Sigmoid)	**0.516**	**0.627**	**0.645**

5.2 Effect of Fusion Mechanism

RQ2: How does the choice of fusion mechanism impact REGENT's performance, and to what extent does its effectiveness depend on a specific fusion strategy versus its dual-pathway architecture? To answer this, we replaced our default learned sigmoid gate with six alternative fusion strategies and retrained our model: (1) **Gated GELU**, employing a sophisticated nonlinear gating network with GELU activation and dropout regularization; (2) **Additive**, using straightforward element-wise addition followed by normalization; (3) **Equal Weighting**, applying a fixed 50-50 average of token and entity outputs; (4) **Learned Tanh**, implementing tanh-based gating as an alternative to sigmoid activation; (5) **Hard Switch**, utilizing a binary selector that favors entity attention when entities are present; and (6) **Attention-based**, leveraging multi-head attention over stacked token and entity representations.

The results reveal a remarkable consistency across these diverse methods, with overall MAP scores clustering within a tight ≈1% margin (0.496 to 0.502, see Table 4). This narrow variance provides powerful evidence that REGENT's effectiveness stems primarily from its dual-pathway architecture itself—which successfully separates lexical and semantic processing—rather than from a specific, highly-tuned fusion technique.

However, a deeper analysis reveals that this macro-level stability masks important specializations. No single fusion method was universally optimal. Instead, different query types and difficulty levels responded best to different strategies, pointing toward the potential of adaptive routing. Our analysis shows that a query-specific fusion selection could yield substantial improvements—up to 41.5% for medium-difficulty queries. The robustness of this difficulty-based analysis is supported by strong correlations between WIG scores and performance (0.67–0.71), validating our classification approach.

Takeaway. The key finding is that REGENT's performance is fundamentally robust to the choice of fusion mechanism, confirming that the dual-pathway design is the core contribution. Furthermore, the fact that different query types favor different fusion

Table 5: Ablation study for entity scoring choices. Results reported on TREC Core 2018.

Method	MAP	nDCG@20	P@20
RENET(BM25)	0.132▼	0.179▼	0.207▼
REGENT(MaxSim)	0.071▼	0.121▼	0.145▼
REGENT(CentroidSim)	0.081▼	0.121▼	0.149▼
REGENT(LogReg)	0.093▼	0.222▼	0.213▼
REGENT(BERT)	**0.516**	**0.627**	**0.645**

strategies suggests a promising and quantifiable avenue for future work in query-adaptive routing, where the fusion method could be selected dynamically based on query characteristics.

5.3 Effect of Supervised Entity Ranker

RQ3: Is the supervised BERT-based entity ranker (Section 3.3) a necessary component for REGENT's performance, or can simpler (unsupervised or less complex supervised) entity scoring methods achieve comparable results? To answer this, we replaced our main entity ranker with four alternatives: (1) **BM25**: An unsupervised lexical approach where candidate entities are scored by running the query against a corpus of their textual descriptions (from DBpedia), (2) **MaxSim**: An unsupervised semantic approach where candidate entities are scored by their maximum (Wikipedia2Vec) cosine similarity to the entities linked to the query, (3) **CentroidSim**: An unsupervised method that scores each entity by its cosine similarity to the average query entity embedding (centroid), and (4) **LogReg**: A simpler supervised approach using a logistic regression classifier instead of a full BERT-based ranker. We then used the entity sets obtained using each of these methods to train and evaluate our full REGENT model.

The results, presented in Table 5, show that replacing our supervised entity ranker with any simpler method leads to a complete collapse in performance. The MAP score plummets from 0.516 to as low as 0.093—a drop of nearly 82%. Neither unsupervised methods (BM25, MaxSim, CentroidSim) nor a simpler supervised model (LogReg) could produce an entity signal useful for REGENT's downstream reasoning. This finding underscores that the model's success is not from using any entity information, but is critically dependent on the high-quality, contextually-aware relevance scores that only the sophisticated, BERT-based ranker can provide. Because these high-quality scores are intrinsically linked to the scaling operation, this result also justifies our focus on evaluating the entire entity pipeline rather than attempting to isolate the scaling effect alone.

Takeaway. The complexity of our supervised entity selection pipeline is not just justified; it is essential. The ability to accurately identify and score the most relevant entities for a given query is a prerequisite for REGENT's success and a core component of its state-of-the-art performance.

5.4 Feature Pipeline vs. Model Design

RQ4: What is the primary driver of REGENT's effectiveness: the supervised entity selection pipeline or its novel architectural design for processing those features? To answer this, we designed a controlled experiment to isolate each contribution. We created **BERT-Entity**, a stronger baseline that receives exactly the same relevance-scored entity features as REGENT, but processes them through a standard cross-encoder architecture via simple feature concatenation. This creates a clear three-way comparison.

The Impact of the Entity Pipeline. First, we measured the effect of our entity selection pipeline on its own. Simply providing these high-quality entity features to a standard BERT model yields substantial improvements, boosting MAP from 0.302 to 0.390 on TREC Core 2018—a 29.1% relative gain. This result confirms that our supervised entity selection pipeline is a highly effective method for representing documents for neural ranking.

The Added Value of the REGENT Architecture. Next, we compared REGENT to the much stronger BERT-Entity baseline. Since both models receive identical inputs, this comparison isolates the architectural advantage. REGENT achieves another significant 32.3% relative improvement in MAP (from 0.390 to 0.516). This demonstrates that *how* signals are processed is as crucial as *what* signals are available. REGENT's relevance-guided attention, which deeply integrates entity and token reasoning, proves architecturally superior to the simpler approach of feature concatenation.

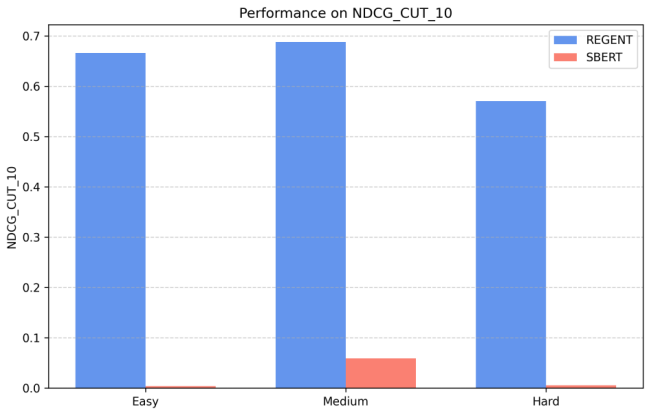
Takeaway. REGENT's state-of-the-art performance emerges from the synergy of two distinct and significant contributions: an effective, supervised approach to entity-aware document representation, and a novel architecture designed to reason with these representations. The results show that while the entity pipeline provides a powerful foundation, the full potential is only realized through REGENT's specialized attention mechanism.

5.5 Analysis by Query Characteristics

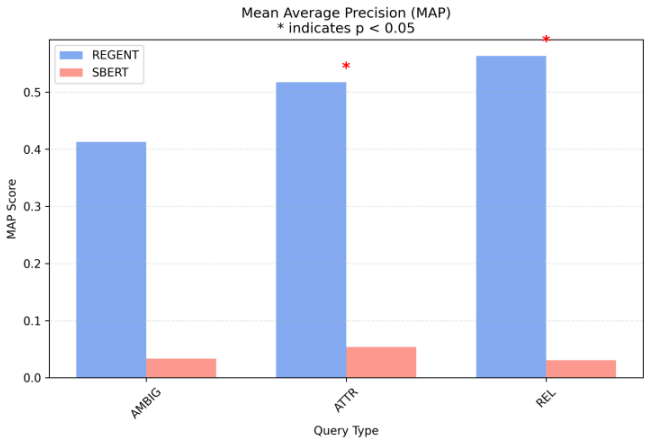
RQ5: How does REGENT's performance vary across different types of query semantic complexity? To answer this, we analyzed results across three query categories: (1) Relational (REL)–queries seeking relationships between entities; (2) Attributive (ATTR)–queries about properties of a single entity; and (3) Ambiguous/Keyword (AMBIG)–broad or definitional queries. We adopted a two-stage annotation process for classifying all 50 topics. First, two state-of-the-art LLMs (Gemini 2.5 Pro and Claude Sonnet 4) independently labeled the queries. Their agreement yielded a Cohen's Kappa of 0.42 ("Moderate Agreement"); 11 of the 12 disagreements occurred between ATTR and REL, indicating a nuanced semantic overlap. In the second stage, the authors adjudicated all labels to establish a gold-standard annotation.

The results provide a clear validation of our hypothesis. REGENT's performance advantage grows precisely as the query's semantic complexity increases. REGENT's gains over SBERT, chosen as a representative and strong sentence-transformer baseline, are largest for REL queries (+0.533 MAP, $n = 10$, $p<0.05$), followed by ATTR (+0.463 MAP, $n = 35$, $p<0.05$), and AMBIG (+0.379 MAP, $n = 5$, $p<0.05$). Improvements are particularly strong for queries straddling the ATTR–REL boundary, where REGENT's joint entity-relationship modeling proves most beneficial.

Takeaway. REGENT excels at modeling entity-centric queries, especially those involving complex relationships. Its performance advantage scales with the semantic richness of the query—confirming that relevance-aware entity modeling is central to its effectiveness.



(a) Performance comparison of REGENT and SBERT, segmented by query difficulty determined via our WIG-based classifier. SBERT's effectiveness diminishes on harder queries while REGENT's remains high. Hence, REGENT's computational overhead is most impactful on the most challenging queries.



(b) MAP of REGENT and a Sentence-BERT baseline, segmented by the query's semantic type. As hypothesized, REGENT's performance advantage grows with the relational complexity of the query, from Ambiguous/Keyword (AMBIG) to Attributive (ATTR) and is most pronounced on Relational (REL) queries that involve interactions between multiple entities.

Figure 2: Query-level analysis on TREC Core 2018.

5.6 Analysis by Query Difficulty

RQ6: How does REGENT's performance vary across different levels of query difficulty, particularly for queries where initial lexical retrieval methods (like BM25) fail? To answer this[3], we first measured each query's difficulty. We define a query's difficulty based on the performance of the initial BM25 ranking (nDCG@20); queries with low initial BM25 scores are considered "hard" (Figure 3). For the most difficult queries (0-5th percentile), where BM25 completely fails to retrieve relevant documents (nDCG@20 ≈ 0), REGENT demonstrates a unique capability. While other strong baselines like ColBERT-v2, selected to represent the multi-vector paradigm, offer only modest improvements, REGENT "rescues" these failed queries, achieving a respectable nDCG@20 of 0.35. This trend is not an anomaly. Across all challenging query bins (0-75th percentile), REGENT consistently establishes a substantial performance gap over all other methods. For instance, in the 5-25th percentile, it more than doubles the nDCG@20 of its closest competitor. As expected, for the easiest queries (95-100th percentile) where lexical signals are already strong, all models perform well. However, it is REGENT's exceptional performance on the most challenging queries that validates its architectural design.

Takeaway. REGENT's advantage is most pronounced when simple lexical matching is insufficient. Its ability to turn failed queries into successful ones provides strong evidence that its relevance-guided attention mechanism, leveraging both entity context and token-level signals, offers a more robust path to understanding relevance than architectures more reliant on lexical overlap.

[3]We use minir-plots for this. See: https://github.com/laura-dietz/minir-plots

5.7 Distribution of Relevant Documents

RQ7: How effectively does REGENT re-structure the initial document ranking by promoting relevant documents, especially highly relevant ones and those missed by lexical models? Even for queries where BM25 struggles, REGENT places 374 relevant documents within the top 10, and successfully surfaces nearly half of all relevant content (1,252 documents) within the top 50. The impact is even more pronounced for the most critical documents. REGENT dramatically improves the average rank of highly relevant documents (NIST grade 2) from 171 to 113. This powerful promotion effect, consistent across all query types, demonstrates that REGENT's architecture is not merely refining the top of the list, but actively identifying and elevating high-value documents that simple lexical models miss entirely.

Takeaway. REGENT's success is not just from improving scores, but from its ability to correct the initial ranking. It consistently finds the "needles in the haystack"—highly relevant documents buried deep in the initial candidate set—and promotes them to the top, which is a key requirement for effective retrieval on complex information needs.

5.8 Computational Cost Analysis

RQ8: What is the computational cost of REGENT compared to baselines, and is this overhead justified by its performance gains, especially for queries of varying difficulty? Although not our primary focus, we compared REGENT's runtime to a SentenceBERT cross-encoder on Core18. A query–document pair is processed by REGENT in 25.5 ms—roughly 3× slower than SBERT (8.5 ms). To assess whether this overhead is justified, we used Weighted Information Gain (WIG) [8], a pre-retrieval query performance predictor, to classify queries into difficulty bins. Unlike

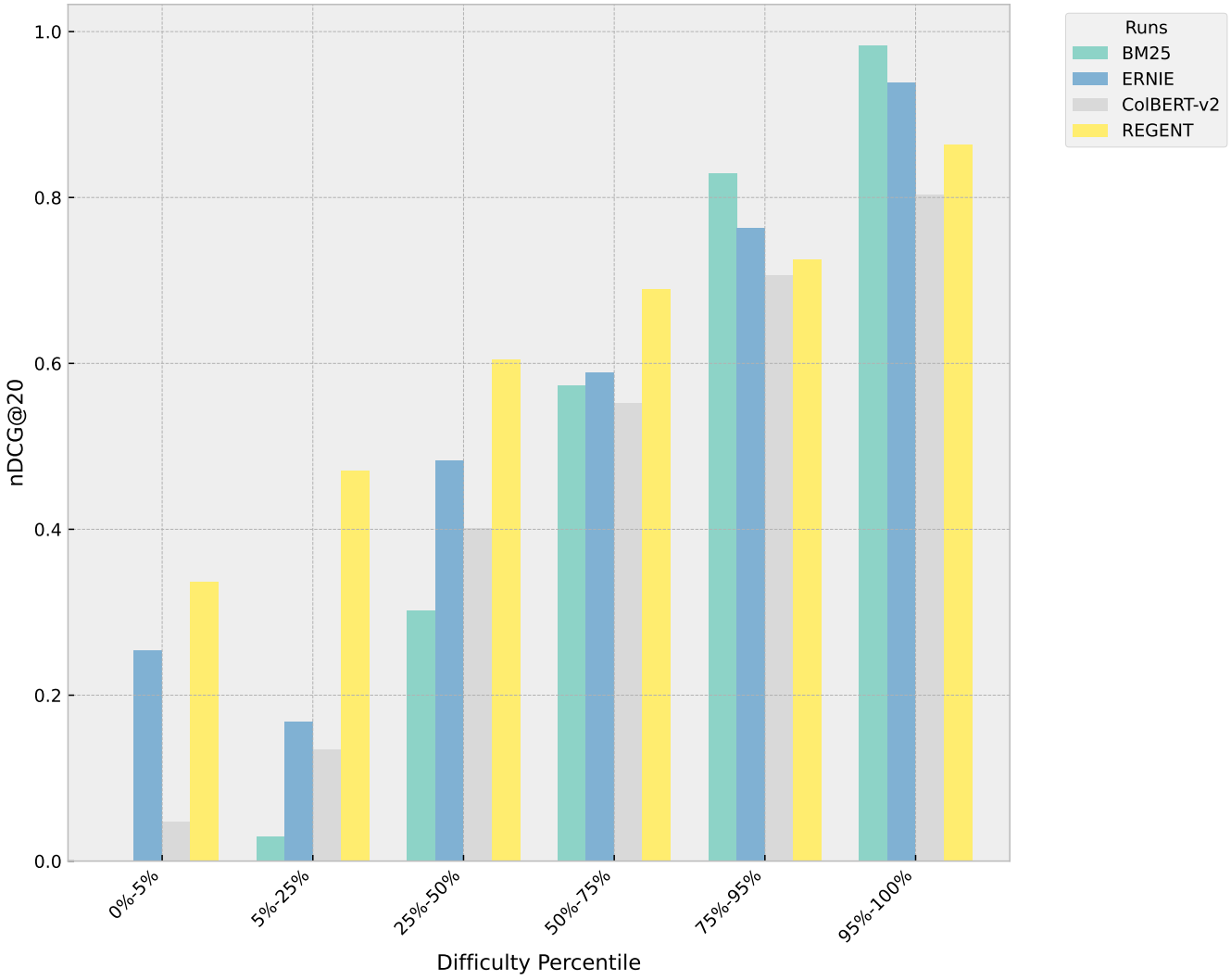


Figure 3: Difficulty test on Core18. 5% most difficult queries for BM25 to the left and the 5% easiest ones to the right. Performance reported as macro-averages across queries.

using the post-retrieval BM25 score, this a priori method is useful for designing adaptive systems that could selectively deploy REGENT on queries predicted to be difficult.

Our results show that REGENT's gains grow sharply with query difficulty. On hard queries, MAP jumps from 0.023 (SBERT) to 0.319 (**+1,316%**); for medium queries, from 0.070 to 0.579 (**+727%**); and even on easy queries, from 0.050 to 0.654. These results (Figure 2a) highlight REGENT's value precisely where traditional models fail.

Takeaway. REGENT's computational cost is a strategic investment. For efficiency-conscious applications, a hybrid approach could deploy REGENT selectively on difficult queries where its order-of-magnitude improvements are most needed. The overhead is justified not despite its cost, but because it enables solving retrieval problems that other methods cannot handle effectively.

5.9 Comparison to Other Entity-Aware Baselines

A key result is REGENT's large performance margin over other entity-aware baselines like EDRM. We attribute this to our *dynamic, query-specific entity processing*. EDRM relies on a static, precomputed entity similarity matrix, treating entity relationships as context-agnostic. In contrast, REGENT's supervised entity ranker and attention mechanism learn to identify and weight entities based on the specific query's context. This ability to dynamically reason about entity relevance appears crucial for handling the complex information needs present in our evaluation datasets

5.10 Case Study Analysis: Sony Cyberattack

To demonstrate how REGENT enables deep semantic reasoning, we examine its performance on a challenging query (QueryID: 342,

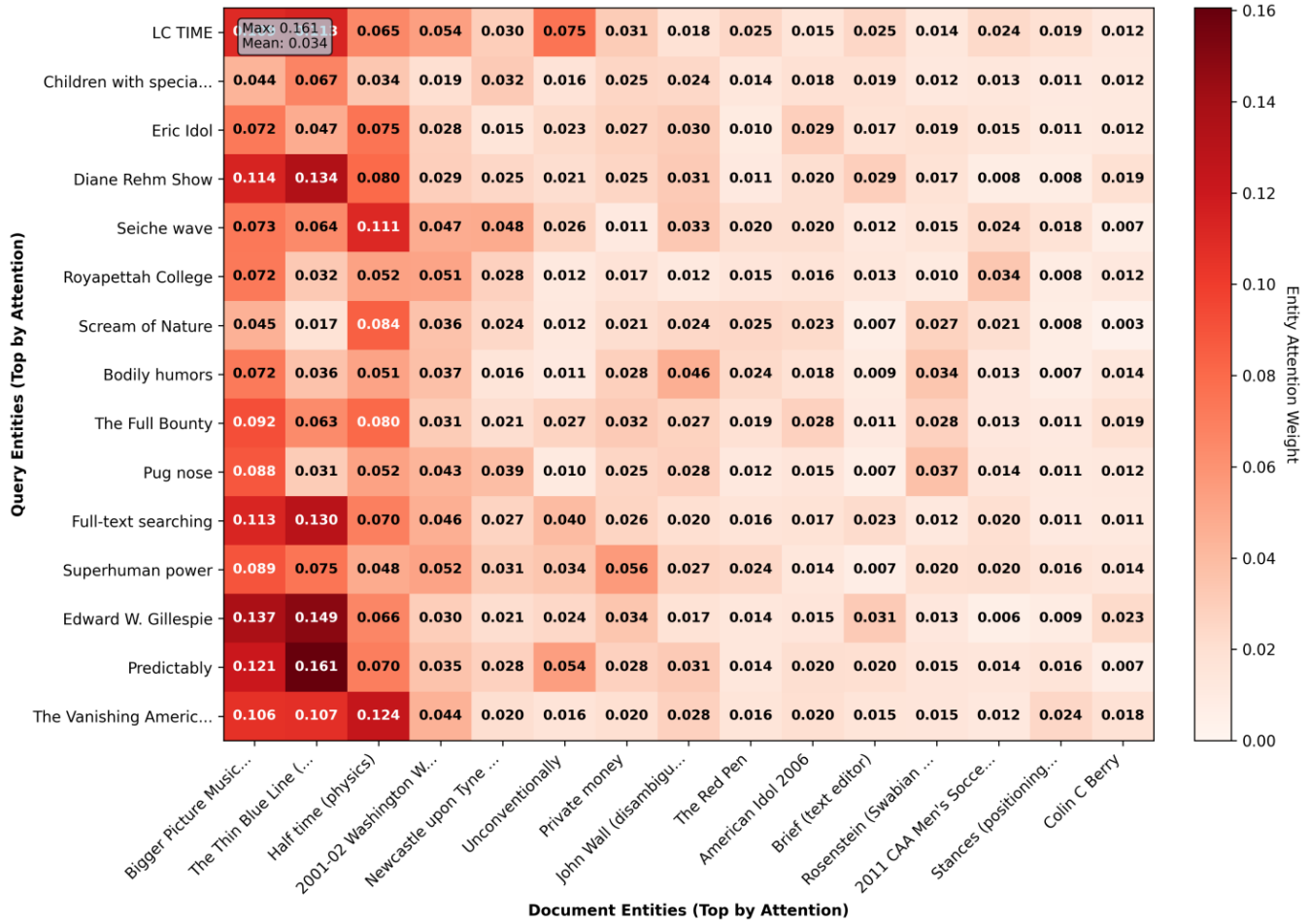


Figure 4: Visualization of entity attention patterns for query "*Sony Cyberattack*".

Type: ATTR) where traditional lexical models fail. The query seeks to identify the specific group responsible for the Sony Pictures cyberattack; generic country-level references are considered non-relevant. A document discussing the White House's response to the WannaCry ransomware receives a BM25 score of 0.0, as it lacks the keyword "cyberattack." REGENT, however, ranks it highly (score: 0.982). Its reasoning begins with entity-to-entity attention, where it finds no direct match for "Sony" but identifies a strong thematic link between query entities like "`Full-text searching`" (a forensic technique) and document entities like "`The Thin Blue Line`" (associated with investigations). As shown in Figure 4, rather than indicating a direct relation, the high attention score reflects a shared thematic context of "Investigation and Attribution," aligning the query's goal —- attributing responsibility—with the document's content on leaks, hacks, and attribution. The context captured by A_e informs the entity-to-token attention module (A_{et}), which highlights key evidence—e.g., linking the "crippling hack on Sony Pictures in 2014" to "Lazarus" despite no exact lexical match. The adaptive fusion mechanism then prioritizes the entity pathway, as weak lexical signals are outweighed by strong semantic cues.

Takeaway. This case illustrates how REGENT surfaces deeply relevant documents that traditional models miss. By constructing a semantic context through entity relationships, it moves beyond simple term matching and captures nuanced aspects of user intent, even when lexical overlap is minimal.

6 CONCLUSION

We present REGENT, a multi-vector neural re-ranking model that advances integration of traditional IR signals and entity knowledge within neural architectures. At its core is *relevance-guided attention*, a novel mechanism that dynamically fuses lexical and semantic cues to guide attention computations, moving beyond uniform token interactions. Across three large-scale datasets featuring long, information-rich documents and complex queries, REGENT delivers state-of-the-art performance, outperforming BM25 by up to 108% and surpassing strong baselines like RankT5, ColBERT, and even LLM-based re-rankers such as RankVicuna. Critically, removing the entity-aware pathway leads to a catastrophic 74% drop in performance, validating our hypothesis that entities serve as a vital "semantic skeleton" for understanding document relevance.

Our work makes three fundamental contributions to neural IR: (1) We demonstrate that token-level BM25 integration significantly outperforms document-level approaches, providing strong evidence that lexical signals must be integrated at the appropriate granularity to be effective in multi-vector attention frameworks. (2) We show that entity information—traditionally confined to single-vector models—can be effectively integrated into multi-vector architectures through our dual-pathway attention design, opening new avenues for entity-aware neural retrieval. (3) We introduce the first ranking-guided attention mechanism that dynamically fuses lexical matching with entity-level semantics, moving beyond static signal combination toward contextual relevance modeling.

REGENT marks a meaningful step toward retrieval systems capable of human-like semantic reasoning. By unifying traditional IR strengths with modern neural architectures, it delivers both strong empirical gains and a solid foundation for future advances in entity-aware neural IR. While we focused on a specific instantiation of relevance-guided attention, future work could explore alternative integration methods (e.g., gating) or attention variants to further build on this paradigm. Our work focuses on English-language retrieval, and adapting the token-level BM25 integration for languages without clear word boundaries, such as CJK, remains an important area for future work.

REFERENCES

[1] Zeynep Akkalyoncu Yilmaz, Wei Yang, Haotian Zhang, and Jimmy Lin. 2019. Cross-Domain Modeling of Sentence-Level Evidence for Document Retrieval. 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)*. Association for Computational Linguistics, Hong Kong, China, 3490–3496. https://doi.org/10.18653/v1/D19-1352

[2] James Allan, Donna Harman, Evangelos Kanoulas, Dan Li, Christophe Van Gysel, and Ellen M Voorhees. 2017. TREC 2017 Common Core Track Overview. In *TREC*.

[3] Arian Askari, Amin Abolghasemi, Gabriella Pasi, Wessel Kraaij, and Suzan Verberne. 2023. Injecting the BM25 Score as Text Improves BERT-Based Re-rankers. In *Advances in Information Retrieval*, Jaap Kamps, Lorraine Goeuriot, Fabio Crestani, Maria Maistro, Hideo Joho, Brian Davis, Cathal Gurrin, Udo Kruschwitz, and Annalina Caputo (Eds.). Springer Nature Switzerland, Cham, 66–83.

[4] Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. MS MARCO: A Human Generated Machine Reading Comprehension Dataset. *arXiv preprint arXiv:1611.09268* (2016).

[5] Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2023. BGE M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation. arXiv:2309.07597 [cs.CL]

[6] Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. *CoRR* abs/2003.10555 (2020). arXiv:2003.10555 https://arxiv.org/abs/2003.10555

[7] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, Ellen M. Voorhees, and Ian Soboroff. 2021. TREC Deep Learning Track: Reusable Test Collections in the Large Data Regime. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, Canada) *(SIGIR '21)*. Association for Computing Machinery, New York, NY, USA, 2369–2375. https://doi.org/10.1145/3404835.3463249

[8] Steve Cronen-Townsend, Yun Zhou, and W. Bruce Croft. 2002. Predicting query performance. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Tampere, Finland) *(SIGIR '02)*. Association for Computing Machinery, New York, NY, USA, 299–306. https://doi.org/10.1145/564376.564429

[9] Zhuyun Dai and Jamie Callan. 2019. Deeper Text Understanding for IR with Contextual Neural Language Modeling. *CoRR* abs/1905.09217 (2019). arXiv:1905.09217 http://arxiv.org/abs/1905.09217

[10] Zhuyun Dai and Jamie Callan. 2020. Context-Aware Term Weighting For First Stage Passage Retrieval. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, China) *(SIGIR '20)*. Association for Computing Machinery, New York, NY, USA, 1533–1536. https://doi.org/10.1145/3397271.3401204

[11] Zhuyun Dai, Vincent Y. Zhao, Ji Ma, Yi Luan, Jianmo Ni, Jing Lu, Anton Bakalov, Kelvin Guu, Keith B. Hall, and Ming-Wei Chang. 2022. Promptagator: Few-shot Dense Retrieval From 8 Examples. arXiv:2209.11755 [cs.CL] https://arxiv.org/abs/2209.11755

[12] Jeffrey Dalton, Laura Dietz, and James Allan. 2014. Entity query feature expansion using knowledge base links. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*. ACM, 365–374.

[13] 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 http://arxiv.org/abs/1810.04805

[14] Laura Dietz, Ben Gamari, Jeff Dalton, and Nick Craswell. 2018. TREC Complex Answer Retrieval Overview.. In *TREC*.

[15] Laura Dietz, Manisha Verma, Filip Radlinski, and Nick Craswell. 2017. TREC Complex Answer Retrieval Overview.. In *Proceedings of Text REtrieval Conference (TREC)*.

[16] Yifan Ding, Amrit Poudel, Qingkai Zeng, Tim Weninger, Balaji Veeramani, and Sanmitra Bhattacharya. 2024. EntGPT: Linking Generative Large Language Models with Knowledge Bases. arXiv:2402.06738 [cs.CL] https://arxiv.org/abs/2402.06738

[17] Faezeh Ensan and Ebrahim Bagheri. 2017. Document Retrieval Model Through Semantic Linking. In *Proceedings of the 10th ACM International Conference on Web Search and Data Mining* (Cambridge, United Kingdom) *(WSDM '17)*. Association for Computing Machinery, New York, NY, USA, 181–190. https://doi.org/10.1145/3018661.3018692

[18] Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. 2021. SPLADE: Sparse Lexical and Expansion Model for First Stage Ranking. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, Canada) *(SIGIR '21)*. Association for Computing Machinery, New York, NY, USA, 2288–2292. https://doi.org/10.1145/3404835.3463098

[19] Evgeniy Gabrilovich and Shaul Markovitch. 2009. Wikipedia-based Semantic Interpretation for Natural Language Processing. *Journal of Artificial Intelligence Research* 34 (2009), 443–498.

[20] Luyu Gao, Zhuyun Dai, Tongfei Chen, Zhen Fan, Benjamin Van Durme, and Jamie Callan. 2021. Complement Lexical Retrieval Model with Semantic Residual Embeddings. In *Advances in Information Retrieval: 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28 – April 1, 2021, Proceedings, Part I*. Springer-Verlag, Berlin, Heidelberg, 146–160. https://doi.org/10.1007/978-3-030-72113-8_10

[21] Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2023. Precise Zero-Shot Dense Retrieval without Relevance Labels. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (Eds.). Association for Computational Linguistics, Toronto, Canada, 1762–1777. https://doi.org/10.18653/v1/2023.acl-long.99

[22] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W. Bruce Croft. 2016. A Deep Relevance Matching Model for Ad-Hoc Retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management* (Indianapolis, Indiana, USA) *(CIKM '16)*. Association for Computing Machinery, New York, NY, USA, 55–64. https://doi.org/10.1145/2983323.2983769

[23] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. DeBERTa: Decoding-enhanced BERT with Disentangled Attention. *CoRR* abs/2006.03654 (2020). arXiv:2006.03654 https://arxiv.org/abs/2006.03654

[24] 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 *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management* (San Francisco, California, USA) *(CIKM '13)*. Association for Computing Machinery, New York, NY, USA, 2333–2338. https://doi.org/10.1145/2505515.2505665

[25] Kai Hui, Andrew Yates, Klaus Berberich, and Gerard de Melo. 2017. PACRR: A Position-Aware Neural IR Model for Relevance Matching. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Copenhagen, Denmark, 1049–1058. https://doi.org/10.18653/v1/D17-1110

[26] Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2020. Poly-encoders: Architectures and Pre-training Strategies for Fast and Accurate Multi-sentence Scoring. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net. https://openreview.net/forum?id=SkxgnnNFvH

[27] Zihang Jiang, Weihao Yu, Daquan Zhou, Yunpeng Chen, Jiashi Feng, and Shuicheng Yan. 2020. ConvBERT: Improving BERT with Span-based Dynamic Convolution. *CoRR* abs/2008.02496 (2020). arXiv:2008.02496 https://arxiv.org/abs/2008.02496

[28] Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. Search-R1: Training LLMs to Reason and Leverage Search Engines with Reinforcement Learning. arXiv:2503.09516 [cs.CL] https://arxiv.org/abs/2503.09516

[29] Chris Kamphuis, Aileen Lin, Siwen Yang, Jimmy Lin, Arjen P. de Vries, and Faegheh Hasibi. 2023. MMEAD: MS MARCO Entity Annotations and Disambiguations. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Taipei, Taiwan) *(SIGIR '23)*. Association for Computing Machinery, New York, NY, USA, 2817–2825. https://doi.org/10.1145/3539618.3591887

[30] Chris Kamphuis, Aileen Lin, Siwen Yang, Jimmy Lin, Arjen P. de Vries, and Faegheh Hasibi. 2023. MMEAD: MS MARCO Entity Annotations and Disambiguations. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Taipei, Taiwan) *(SIGIR '23)*. Association for Computing Machinery, New York, NY, USA, 2817–2825. https://doi.org/10.1145/3539618.3591887

[31] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Online, 6769–6781. https://doi.org/10.18653/v1/2020.emnlp-main.550

[32] 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* (Virtual Event, China) *(SIGIR '20)*. Association for Computing Machinery, New York, NY, USA, 39–48. https://doi.org/10.1145/3397271.3401075

[33] Diederik P Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980* (2014).

[34] Carlos Lassance, Hervé Déjean, Thibault Formal, and Stéphane Clinchant. 2024. SPLADE-v3: New Baselines for SPLADE. arXiv:2403.06789 [cs.IR] https://arxiv.org/abs/2403.06789

[35] Canjia Li, Yingfei Sun, Ben He, Le Wang, Kai Hui, Andrew Yates, Le Sun, and Jungang Xu. 2018. NPRF: A Neural Pseudo Relevance Feedback Framework for Ad-hoc Information Retrieval. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Brussels, Belgium, 4482–4491. https://doi.org/10.18653/v1/D18-1478

[36] Canjia Li, Andrew Yates, Sean MacAvaney, Ben He, and Yingfei Sun. 2020. PARADE: Passage Representation Aggregation for Document Reranking. *CoRR* abs/2008.09093 (2020). arXiv:2008.09093 https://arxiv.org/abs/2008.09093

[37] Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. 2020. Distilling Dense Representations for Ranking using Tightly-Coupled Teachers. *CoRR* abs/2010.11386 (2020). arXiv:2010.11386 https://arxiv.org/abs/2010.11386

[38] Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. 2021. In-Batch Negatives for Knowledge Distillation with Tightly-Coupled Teachers for Dense Retrieval. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, Anna Rogers, Iacer Calixto, Ivan Vulić, Naomi Saphra, Nora Kassner, Oana-Maria Camburu, Trapit Bansal, and Vered Shwartz (Eds.). Association for Computational Linguistics, Online, 163–173. https://doi.org/10.18653/v1/2021.repl4nlp-1.17

[39] Xitong Liu and Hui Fang. 2015. Latent entity space: a novel retrieval approach for entity-bearing queries. *Information Retrieval Journal* 18, 6 (2015), 473–503.

[40] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR* abs/1907.11692 (2019). arXiv:1907.11692 http://arxiv.org/abs/1907.11692

[41] Zhenghao Liu, Chenyan Xiong, Maosong Sun, and Zhiyuan Liu. 2018. Entity-Duet Neural Ranking: Understanding the Role of Knowledge Graph Semantics in Neural Information Retrieval. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Melbourne, Australia, 2395–2405. https://doi.org/10.18653/v1/P18-1223

[42] Xueguang Ma, Kai Sun, Ronak Pradeep, and Jimmy Lin. 2021. A Replication Study of Dense Passage Retriever. *CoRR* abs/2104.05740 (2021). arXiv:2104.05740 https://arxiv.org/abs/2104.05740

[43] Iain Mackie, Paul Owoicho, Carlos Gemmell, Sophie Fischer, Sean MacAvaney, and Jeffrey Dalton. 2022. CODEC: Complex Document and Entity Collection. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Madrid, Spain) *(SIGIR '22)*. Association for Computing Machinery, New York, NY, USA, 3067–3077. https://doi.org/10.1145/3477495.3531712

[44] Bhaskar Mitra and Nick Craswell. 2019. An Updated Duet Model for Passage Re-ranking. *CoRR* abs/1903.07666 (2019). arXiv:1903.07666 http://arxiv.org/abs/1903.07666

[45] Shahrzad Naseri, Jeffrey Dalton, Andrew Yates, and James Allan. 2021. CEQE: Contextualized Embeddings for Query Expansion. In *Advances in Information Retrieval: 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28 – April 1, 2021, Proceedings, Part I*. Springer-Verlag, Berlin, Heidelberg, 467–482. https://doi.org/10.1007/978-3-030-72113-8_31

[46] Thong Nguyen, Shubham Chatterjee, Sean MacAvaney, Iain Mackie, Jeff Dalton, and Andrew Yates. 2024. DyVo: Dynamic Vocabularies for Learned Sparse Retrieval with Entities. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (Eds.). Association for Computational Linguistics, Miami, Florida, USA, 767–783. https://doi.org/10.18653/v1/2024.emnlp-main.45

[47] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. *CoRR* abs/1611.09268 (2016). arXiv:1611.09268 http://arxiv.org/abs/1611.09268

[48] Rodrigo Frassetto Nogueira and Kyunghyun Cho. 2019. Passage Re-ranking with BERT. *CoRR* abs/1901.04085 (2019). arXiv:1901.04085 http://arxiv.org/abs/1901.04085

[49] Rodrigo Frassetto Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019. Document Expansion by Query Prediction. *CoRR* abs/1904.08375 (2019). arXiv:1904.08375 http://arxiv.org/abs/1904.08375

[50] Francesco Piccinno and Paolo Ferragina. 2014. From TagME to WAT: A New Entity Annotator. In *Proceedings of the First International Workshop on Entity Recognition & Disambiguation* (Gold Coast, Queensland, Australia) *(ERD '14)*. Association for Computing Machinery, New York, NY, USA, 55–62. https://doi.org/10.1145/2633211.2634350

[51] Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. 2023. RankVicuna: Zero-Shot Listwise Document Reranking with Open-Source Large Language Models. arXiv:2309.15088 [cs.IR] https://arxiv.org/abs/2309.15088

[52] Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. 2023. RankZephyr: Effective and Robust Zero-Shot Listwise Reranking is a Breeze! arXiv:2312.02724 [cs.IR] https://arxiv.org/abs/2312.02724

[53] Hadas Raviv, Oren Kurland, and David Carmel. 2016. Document Retrieval Using Entity-Based Language Models. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Pisa, Italy) *(SIGIR '16)*. Association for Computing Machinery, New York, NY, USA, 65–74. https://doi.org/10.1145/2911451.2911508

[54] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *CoRR* abs/1908.10084 (2019). arXiv:1908.10084 http://arxiv.org/abs/1908.10084

[55] Tao Shen, Xiubo Geng, Chongyang Tao, Can Xu, Xiaolong Huang, Binxing Jiao, Linjun Yang, and Daxin Jiang. 2023. LexMAE: Lexicon-Bottlenecked Pretraining for Large-Scale Retrieval. arXiv:2208.14754 [cs.IR] https://arxiv.org/abs/2208.14754

[56] Wei Shen, Jianyong Wang, and Jiawei Han. 2015. Entity Linking with a Knowledge Base: Issues, Techniques, and Solutions. *IEEE Transactions on Knowledge and Data Engineering* 27, 2 (2015), 443–460. https://doi.org/10.1109/TKDE.2014.2327028

[57] Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2023. One Embedder, Any Task: Instruction-Finetuned Text Embeddings. In *Findings of the Association for Computational Linguistics: ACL 2023*, Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (Eds.). Association for Computational Linguistics, Toronto, Canada, 1102–1121. https://doi.org/10.18653/v1/2023.findings-acl.71

[58] Daniel Vollmers, Hamada Zahera, Diego Moussallem, and Axel-Cyrille Ngonga Ngomo. 2025. Contextual Augmentation for Entity Linking using Large Language Models. In *Proceedings of the 31st International Conference on Computational Linguistics*, Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert (Eds.). Association for Computational Linguistics, Abu Dhabi, UAE, 8535–8545. https://aclanthology.org/2025.coling-main.570/

[59] Ellen M Voorhees et al. 2003. Overview of the TREC 2003 robust retrieval track.. In *Trec*. 69–77.

[60] Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2024. Text Embeddings by Weakly-Supervised Contrastive Pre-training. arXiv:2212.03533 [cs.CL] https://arxiv.org/abs/2212.03533

[61] Shuai Wang, Shengyao Zhuang, and Guido Zuccon. 2021. BERT-based Dense Retrievers Require Interpolation with BM25 for Effective Passage Retrieval. In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval* (Virtual Event, Canada) *(ICTIR '21)*. Association for Computing Machinery, New York, NY, USA, 317–324. https://doi.org/10.1145/3471158.3472233

[62] Xiao Wang, Craig MacDonald, Nicola Tonellotto, and Iadh Ounis. 2023. ColBERT-PRF: Semantic Pseudo-Relevance Feedback for Dense Passage and Document Retrieval. *ACM Trans. Web* 17, 1, Article 3 (jan 2023), 39 pages. https://doi.org/10.1145/3572405

[63] 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*, Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (Eds.). Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 538–548. https://doi.org/10.18653/v1/2022.emnlp-main.35

[64] Amy Xin, Yunjia Qi, Zijun Yao, Fangwei Zhu, Kaisheng Zeng, Xu Bin, Lei Hou, and Juanzi Li. 2024. LLMAEL: Large Language Models are Good Context Augmenters for Entity Linking. arXiv:2407.04020 [cs.CL] https://arxiv.org/abs/2407.04020

[65] Chenyan Xiong, Jamie Callan, and Tie-Yan Liu. 2017. Word-Entity Duet Representations for Document Ranking. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Shinjuku,

Tokyo, Japan) *(SIGIR '17)*. Association for Computing Machinery, New York, NY, USA, 763–772. https://doi.org/10.1145/3077136.3080768

[66] Chenyan Xiong, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, and Russell Power. 2017. End-to-End Neural Ad-Hoc Ranking with Kernel Pooling. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Shinjuku, Tokyo, Japan) *(SIGIR '17)*. Association for Computing Machinery, New York, NY, USA, 55–64. https://doi.org/10.1145/3077136.3080809

[67] Chenyan Xiong, Russell Power, and Jamie Callan. 2017. Explicit Semantic Ranking for Academic Search via Knowledge Graph Embedding. In *Proceedings of the 26th International Conference on World Wide Web* (Perth, Australia) *(WWW '17)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 1271–1279. https://doi.org/10.1145/3038912.3052558

[68] Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate Nearest Neighbor Negative Contrastive Learning for Dense Text Retrieval. *CoRR* abs/2007.00808 (2020). arXiv:2007.00808 https://arxiv.org/abs/2007.00808

[69] Ikuya Yamada, Akari Asai, Jin Sakuma, Hiroyuki Shindo, Hideaki Takeda, Yoshiyasu Takefuji, and Yuji Matsumoto. 2020. Wikipedia2Vec: An Efficient Toolkit for Learning and Visualizing the Embeddings of Words and Entities from Wikipedia. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, Online, 23–30. https://doi.org/10.18653/v1/2020.emnlp-demos.4

[70] HongChien Yu, Chenyan Xiong, and Jamie Callan. 2021. Improving Query Representations for Dense Retrieval with Pseudo Relevance Feedback. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management* (Virtual Event, Queensland, Australia) *(CIKM '21)*. Association for Computing Machinery, New York, NY, USA, 3592–3596. https://doi.org/10.1145/3459637.3482124

[71] Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced Language Representation with Informative Entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 1441–1451. https://doi.org/10.18653/v1/P19-1139

[72] Zhi Zheng, Kai Hui, Ben He, Xianpei Han, Le Sun, and Andrew Yates. 2020. BERT-QE: Contextualized Query Expansion for Document Re-ranking. In *Findings of the Association for Computational Linguistics: EMNLP 2020*. Association for Computational Linguistics, Online, 4718–4728. https://doi.org/10.18653/v1/2020.findings-emnlp.424

[73] Honglei Zhuang, Zhen Qin, Rolf Jagerman, Kai Hui, Ji Ma, Jing Lu, Jianmo Ni, Xuanhui Wang, and Michael Bendersky. 2023. RankT5: Fine-Tuning T5 for Text Ranking with Ranking Losses. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Taipei, Taiwan) *(SIGIR '23)*. Association for Computing Machinery, New York, NY, USA, 2308–2313. https://doi.org/10.1145/3539618.3592047

[74] Shengyao Zhuang, Xueguang Ma, Bevan Koopman, Jimmy Lin, and Guido Zuccon. 2025. Rank-R1: Enhancing Reasoning in LLM-based Document Rerankers via Reinforcement Learning. arXiv:2503.06034 [cs.IR] https://arxiv.org/abs/2503.06034