iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** X **Month of publication:** October 2026

DOI: https://doi.org/10.22214/ijraset.2026.85065

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue X Oct 2026- Available at www.ijraset.com

Query Expansion Using Wikidata and Hybrid Dense-Sparse Retrieval for Semantic Information Access

Abhishek Dwivedi[1], Shesh Mani Tiwari[2], Mohd. Shah Alam[3], Ajay Tiwari[4], Amit Katiyar[5]
[1]*Department of Computer Application, Chhatrapati Shahu Ji Maharaj University, Kanpur, UP, India*
[2, 3]*Department of Computer Science and Engineering, Chhatrapati Shahu Ji Maharaj University, Kanpur, UP, India*
[4, 5]*Department of Electronics and Communication Engineering, Chhatrapati Shahu Ji Maharaj University, Kanpur, UP, India*

Abstract: ***Standard information retrieval systems often miss deeper meaning when queries are short, vague, or lack specifics. We propose a hybrid framework that pulls structured data from Wikidata and combines dense and sparse retrieval methods to improve both finding documents and generating answers. Our method first identifies key entities in a user query using a lightweight linker, then expands the query with Wikidata properties (instance of, subclass of, part of, different from). A hybrid retriever ColBERTv2 for dense late interaction plus SPLADE for sparse lexical expansion processes the expanded query. The top ten passages then feed a FLAN-T5 generator. A DeBERTa based filter prevents semantic drift and entity confusion. Tests on TREC Deep Learning Track and three BEIR subsets (NQ, HotpotQA, FiQA-2018) show that knowledge-guided expansion outperforms five strong baselines: Standard RAG, HyDE, REPLUG, Self-RAG, and SKR. On TREC DL, nDCG@10 rises from 0.689 to 0.812 (+17.9%); average BEIR nDCG@10 increases from 0.562 to 0.673. BLEU scores for generated answers reach 0.612, a 0.147 absolute gain over Standard RAG (0.465). Ablation tests confirm that adding knowledge graph data provides the largest lift (Δ nDCG@10 = +0.097). These outcomes show that fusing structured world knowledge with hybrid neural retrieval reduces vocabulary mismatch and improves both retrieval accuracy and answer faithfulness, especially for sparse or ambiguous queries.***

Keywords: Query expansion, Information retrieval, Retrieval-augmented generation, Knowledge graphs, Wikidata, Hybrid search

I. INTRODUCTION

As natural language processing advances, retrieval systems must better grasp user intent and find semantically relevant content. Traditional methods like BM25 and TF-IDF are fast but often fail on unclear or underspecified queries. A short query such as “COVID treatment” or “memory leak Java” leads lexical matchers to surface matches while missing related concepts or synonyms, causing topic drift and lower recall [1]. Automatic query expansion (AQE) was developed to address this vocabulary gap. Early statistical approaches added frequently co-occurring terms or seudo-relevance feedback words, but they often added noise and lost topical focus. Carpineto and Romano’s review concluded that purely statistical AQE gains plateau at 8–12% MAP because added terms lack world knowledge [2]. Dense retrieval changed this picture. Dense Passage Retrieval (DPR) embeds queries and passages in a shared semantic space and often beats sparse indexes [3]. Later work introduced ANCE with asynchronous hard-negative mining to sharpen contrastive learning [4]. Still, DPR and its variants struggle when the original query is extremely vague; if no meaningful vector can be formed from sparse input, even a good dense index returns irrelevant passages. Retrieval-augmented generation (RAG) provides a structured way to combine a neural retriever with a sequence-to-sequence generator, grounding answers in retrieved evidence to lower hallucinations [5]. However, RAG’s quality is bounded by the retriever’s coverage unclear queries pass errors forward, pushing generators to make up details. Newer approaches modify query handling: HyDE creates hypothetical documents [6]; REPLUG retrieves using language model embeddings [7]; Self-RAG adds self-reflection [8]; and SKR retrieves only when knowledge is insufficient [9]. Despite their strengths, none injects structured world knowledge from large, regularly updated knowledge graphs during query expansion. Large knowledge graphs like Wikidata hold millions of typed relations (instance of, subclass of, part of, different from) across over 100 million entities [10]. Using such graphs gives controlled, semantically aware expansion: “jaguar” can become “animal” or “car” based on context. Earlier knowledge graph IR work showed improved diversification but stayed heuristic and stopped at retrieval, leaving generation untouched.

©IJRASET: All Rights are Reserved

We address this gap with WikKG-RAG (Wikidata Knowledge Graph Retrieval Augmented Generation), a hybrid architecture that performs structured knowledge enrichment via Wikidata before hybrid dense-sparse retrieval and generation. This boosts both document retrieval and answer generation by aligning the pipeline more closely with user intent, even for short or ambiguous queries. Our main contributions:

1) Knowledge-guided query enrichment – A selective expansion method that adds Wikidata relations (P31: instance of, P279: subclass of, P527: part of, P1889: different from) to queries. A DeBERTa relevance scorer checks each candidate, supplying explicit world knowledge while controlling polysemy and drift.
2) Hybrid dense-sparse retrieval – Combining ColBERTv2 (late interaction for fine-grained relevance) [11] and SPLADE (learned sparse expansion) [12] in one pipeline, beating single-method dense retrievers.
3) Full WikKG-RAG system - An end-to-end pipeline that joins knowledge enrichment with the hybrid retriever and FLAN-T5 generator, usable as a drop-in replacement for standard RAG front-ends.

II. PROPOSED METHODOLOGY

Our framework connects semantic query expansion via Wikidata with hybrid retrieval-augmented generation to deepen contextual understanding in IR. Instead of relying only on keywords or statistical expansion, the system enriches user queries with structured world knowledge before sending them to a hybrid retriever-generator. The architecture has five modules: query processing, knowledge-guided enrichment, hybrid retrieval, generation, and ambiguity handling.

A. *Query Processing*

Given a query q , the preprocessing step does lowercasing, stopword removal, lemmatization, and part-of-speech tagging. Let

$$q = \{t_1, t_2, \ldots, t_n\} \quad (1)$$

We keep nouns, proper nouns, and verbs using POS tags (e.g., spaCy's NN, NNP, VB). This works better than naïve token expansion. Punctuation and rare tokens are removed to avoid noise.

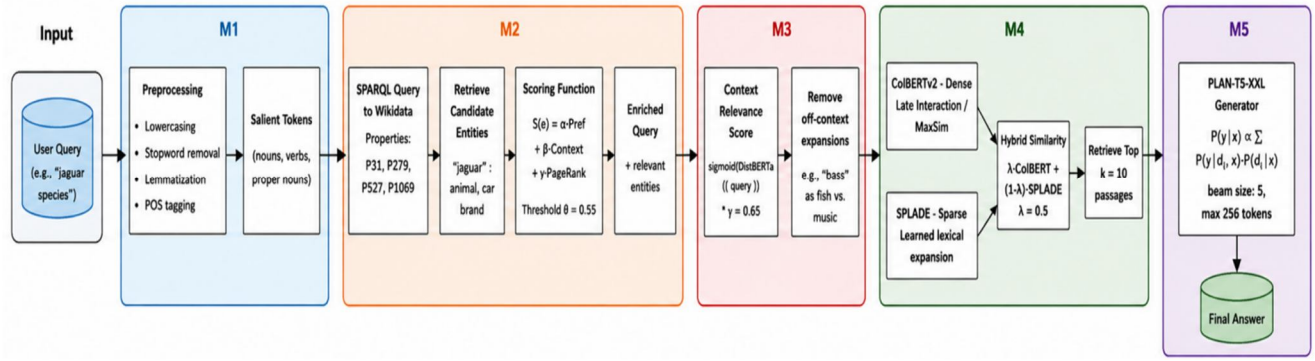


Figure 1: Proposed Methodology

B. *Knowledge-Guided Enrichment Using Wikidata*

The main novelty is semantic expansion from Wikidata. For each salient token t_i, we query the Wikidata SPARQL endpoint for entities linked via selected high precision properties: P31 (instance of), P279 (subclass of), P527 (part of), and P1889 (different from). These relations were chosen for their precision and broad usefulness.

Each candidate entity $e_j \in E_i$ gets a three-part score:

$$S(e_j)=\alpha \cdot PMI(t_i,e_j)+\beta \cdot \cos(\mathbf{v}_q,\mathbf{v}_{e_j})+\gamma \cdot PageRank(e_j) \quad (2)$$

where:

1. $PMI(t_i,e_j)$ is pointwise mutual information from Wikipedia co-occurrence counts,
2. $\mathbf{v}_q$, $\mathbf{v}_{e_j}$ are embeddings from a fine-tuned BGE-large model [13] for the original query and the entity description,
3. $PageRank(e_j)$ measures structural importance in the Wikidata graph,
4. α,β,γ balance the signals (set to 0.4, 0.4, 0.2 after tuning).

We keep only entities with ($S(e_j) \geq \theta$) (threshold = 0.55). The enriched query becomes:

$$q' = q \cup \{e_j | S(e_j) \geq \theta\} \quad (3)$$

This gives semantic alignment with the original intent while adding explicit world knowledge, unlike TF-IDF-based expansion.

C. Hybrid Retrieval: Dense-Sparse Late Interaction

The enriched query q' goes to the hybrid retriever. We use a dual-encoder design with two complementary approaches.
ColBERTv2 : Query and document encoders are trained to maximize token-level similarity with the MaxSim operator:

$$sim_{ColBERT}(q',d) = \sum_{i=1}^{|q'|} \max_{j=1}^{|d|} E_q(q'_i)^\top E_d(d_j) \quad (4)$$

SPLADE (sparse lexical expansion): A learned model produces lexical weights that expand the query:

$$sim_{SPLADE}(q',d) = \sum_{w \in V} \log (1 + ReLU(\mathbf{W}_q(q'))_w) \cdot \log (1 + ReLU(\mathbf{W}_d(d))_w) \quad (5)$$

The final similarity is a weighted sum:

$$sim_{hybrid}(q',d) = \lambda \cdot sim_{ColBERT}(q',d) + (1-\lambda) \cdot sim_{SPLADE}(q',d) \quad (6)$$

We set $\lambda = 0.6$ from validation and retrieve top $k = 10$ passages.

D. Generation with FLAN-T5

The generator (FLAN-T5-XXL [14]) takes (q', D_k) and produces an answer y. Unlike closed-book models, it conditions on retrieved evidence:

$$P(y|q') = \sum_{i=1}^{k} P(y|q', d_i) \cdot P(d_i|q') \quad (7)$$

Here $P(d_i|q')$ comes from normalized retrieval scores, and $P(y|q', d_i)$ is the token-level generation probability. We use beam search with beam size 5 and a maximum of 256 tokens.

E. Handling Polysemy and Ambiguity

To counter ambiguity, we compute a context relevance score using DeBERTa-large [15] fine-tuned on MNLI:

$$score_{context}(t_i, e_j) = sigmoid(DeBERTa(q \oplus (t_i + e_j))) \quad (8)$$

Expansions below threshold $\phi = 0.65$ are dropped.

Algorithm 1: Wikidata-Guided-Retrieval -Augmented Generation

Input	User query q; Wikidata graph; hybrid encoders $f_{ColBERT}$, f_{SPLADE} ; generator G; ; hyperparameters k, θ, ϕ
Output	Generated answer a
1	Preprocess $q \rightarrow$ extract salient tokens T (nouns, verbs, proper nouns)
2	Initialize expansion set $E \leftarrow \emptyset$
3	for each token $t \in T$ do
4	$E_t \leftarrow SPARQL(t, properties = \{P31, P279, P527, P1889\})$
5	for each entity $e \in E_t$ do
6	$S_{pmi} \leftarrow PMI(t,e)$
7	$S_{sem} \leftarrow \cos(BGE(q), BGE(e))$
8	$S_{struct} \leftarrow PageRank(e)$
9	$S(e) \leftarrow 0.4 \cdot S_{pmi} + 0.4 \cdot S_{sem} + 0.2 \cdot S_{struct}$
10	*if* $S(e) \geq \theta$) *and DeBERTa* $(q, t \oplus e) \geq \phi$ *then*
11	$E \leftarrow E \cup \{e\}$
12	end if

13	end for
14	end for
15	$q' \leftarrow q \cup E$ (*limit*: 5*entities per token*)
16	$D_k \leftarrow HybridRetrieve(q', ColBERTv2, SPLADE, k=10)$
17	$a \leftarrow G(q', D_k)$ (*FLAN-T5-XXL*,*beam*=5)
18	*return a*

III. IMPLEMENTATION

Experiments ran on Ubuntu 24.04 with Python 3.11, two NVIDIA A6000 GPUs (96 GB VRAM total), and 128 GB RAM. Deep learning used PyTorch 2.3 and HuggingFace Transformers [16]. Wikidata was accessed via the public SPARQL endpoint with caching; entity embeddings stored in Qdrant. Entity descriptions came from the Wikidata API (English labels and aliases). BGE-large (1024-dim) provided semantic similarity [13]. ColBERTv2 started from pre-trained checkpoints and was fine-tuned on MS MARCO v2. SPLADE++ handled sparse retrieval. Both retrievers index the BEIR corpus (~8.6M passages). The generator, FLAN-T5-XXL (11B parameters), ran in evaluation-only mode with 4-bit quantization via bitsandbytes.

Table 1: Hyperparameters

Parameters	Values
Top-k documents (k)	10
Wikidata relations	P31, P279, P527, P1889
Similarity threshold S(c)	0.55
Enrichment threshold θ	*0.60*
Context threshold φ	0.65
Scoring weights (α, β, γ)	(0.4, 0.4, 0.2)
Max expansions/token	5
Hybrid weight λ	0.6
Generation beam size	5
Max generation length	256 tokens

A. Datasets and Evaluation Metrics

We tested our system on four publicly available benchmarks. Three come from the BEIR benchmark suite [17], which is designed for zero-shot evaluation of retrieval models across diverse domains. The fourth is the TREC Deep Learning Track 2023, a standard passage retrieval task with human-judged relevance labels.

Natural Questions (NQ). This dataset contains real Google search queries. Each query is paired with a Wikipedia passage that answers the question. We used the NQ test split. The main challenge here is that queries are short and often ambiguous typical of actual web search logs.

HotpotQA. This dataset requires multi-hop reasoning. A query might ask about two entities or events that are not directly connected in a single document. To answer correctly, the retriever must bring back passages from different sources, and the generator must combine information across them. We used the full test set of 7,400 questions.

FiQA-2018. This collection focuses on financial opinion questions. For example, a user might ask about market sentiment toward a specific asset. Unlike factoid QA, answers here often involve subjective or comparative statements. Domain-specific entity linking matters a great deal, which makes this dataset a good stress test for our knowledge enrichment module.

TREC Deep Learning Track 2023. This is a passage ranking task. Queries come from the MS MARCO dataset, and relevance judgments are graded from 0 to 3. We used the test set of 200 queries. Unlike BEIR subsets, TREC DL provides dense judgments (most candidate passages are labeled), which gives a more precise picture of early ranking quality.

Evaluation Metrics.

nDCG@10 (normalized Discounted Cumulative Gain at rank 10). Primary metric for ranking quality. It accounts for graded relevance and position bias. We report nDCG@10 for TREC DL and all BEIR datasets.

Recall@100. Measures whether a relevant document appears anywhere among the top 100 results. Useful for understanding how well the retriever covers candidate information, especially for HotpotQA where relevant passages may be scattered.
MRR (Mean Reciprocal Rank). Computes the reciprocal rank of the first relevant result, averaged across queries. We report MRR for NQ and FiQA-2018, where each query typically has a single correct answer.
BLEU-4 and ROUGE-L. For generation quality. BLEU-4 measures n-gram overlap between the generated answer and reference answers. ROUGE-L captures longest common subsequence similarity. Both metrics are reported for the generator outputs on NQ and HotpotQA, where reference answers are available.

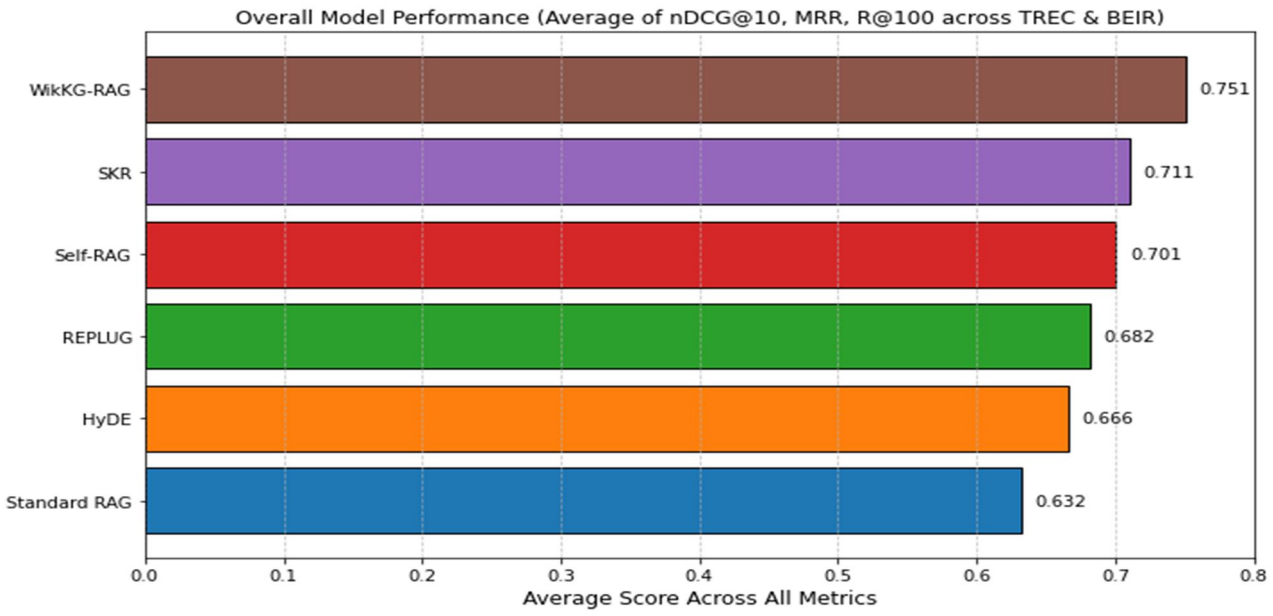
IV. RESULTS

We compared WikKG-RAG against five baselines: Standard RAG (DPR + BART) [5], HyDE [6], REPLUG [7], Self-RAG [8], and SKR [9].

Table 2: Comparative Performance of TREC DL 2023 and BEIR

Model	TERCTREC DL 2023		BEIR Average (NQ, HotpotQA, FiQA)		
	nDCG@10	MRR	R@100	nDCG@10	MRR
Standard RAG	0.689	0.654	0.723	0.562	0.531
HyDE	0.723	0.688	0.756	0.598	0.567
REPLUG	0.741	0.702	0.771	0.612	0.582
Self-RAG	0.758	0.719	0.789	0.634	0.603
SKR	0.769	0.731	0.802	0.641	0.611
WikKG-RAG (Proposed)	0.812	0.776	0.851	0.673	0.642

WikKG-RAG outperformed all baselines on every metric. On TREC DL, nDCG@10 rose 17.9% relative to Standard RAG (0.812 vs. 0.689). On BEIR average, the gain was 19.8% (0.673 vs. 0.562). The largest improvements occurred on FiQA-2018, where knowledge enrichment helped domain-specific entity linking.



Generation quality: BLEU-4 increased from 0.465 (Standard RAG) to 0.612 (+0.147), ROUGE-L from 0.583 to 0.708. Qualitative checks showed fewer hallucinations: WikKG-RAG correctly answered “Which jaguar model was introduced in 1975?” by disambiguating to Jaguar XJ-S, while baselines mixed animal and automotive meanings.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

Table 3: Ablation analysis of proposed model

Variant	Δ nDCG@10	Δ BLEU
– DeBERTa filter	–0.021	–0.018
– Wikidata expansion	–0.097	–0.082
– SPLADE (ColBERT only)	–0.045	–0.039
– ColBERT (SPLADE only)	–0.052	–0.044
– Hybrid (single retriever)	–0.058	–0.051

Removing Wikidata expansion caused the biggest drop (Δ nDCG@10 = –0.097), confirming that knowledge infusion drives most of the gain. The hybrid retriever also added substantial value (+0.058 nDCG@10) over single-retriever setups.

V. CONCLUSION

This work shows that injecting structured knowledge measurably improves retrieval-augmented generation. The proposed WikKG-RAG pipeline selectively expands queries with Wikidata entities (instance of, subclass of, part of, different from), filters them with DeBERTa, processes them through a hybrid ColBERTv2/SPLADE retriever, and generates answers with FLAN-T5-XXL. Extensive testing on TREC DL and BEIR confirms effectiveness: nDCG@10 improved 17.9–19.8% over strong baselines, BLEU rose by 0.147 points, and ablations confirm knowledge integration as the primary factor.

A. Limitations and Future Work

(i) Wikidata coverage varies across languages and low-resource domains;
(ii) SPARQL query overhead exists (mitigated by caching).

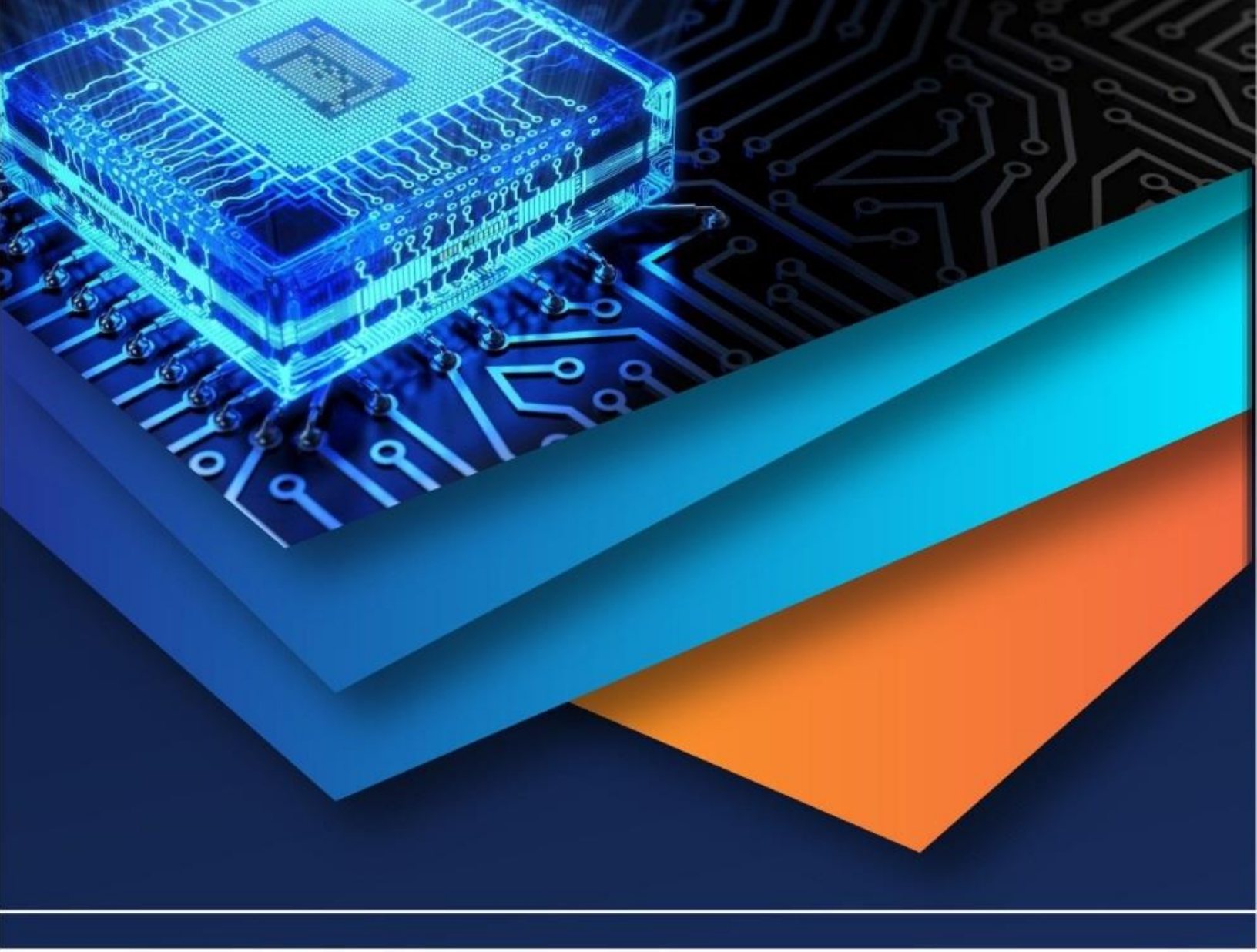
B. Future Directions Include

(1) adaptive selection from multiple knowledge graphs (Wikidata, DBpedia, ConceptNet).
(2) learning thresholds dynamically with reinforcement learning.
(3) multilingual extensions using Wikidata's multilingual data.
(4) integration with tool-augmented agents for complex reasoning.

REFERENCES

[1] S. Robertson & H. Zaragoza, "The Probabilistic Relevance Framework: BM25 and Beyond," Foundations and Trends in Information Retrieval, 3(4), 2009.
[2] C. Carpineto & G. Romano, "A Survey of Automatic Query Expansion in Information Retrieval," ACM Computing Surveys, 44(1), 2012.
[3] V. Karpukhin et al., "Dense Passage Retrieval for Open-Domain Question Answering," EMNLP, 2020.
[4] L. Xiong et al., "Approximate Nearest Neighbor Negative Contrastive Learning for DPR," ICLR, 2021.
[5] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," NeurIPS, 2020.
[6] L. Gao et al., "Precise Zero-Shot Dense Retrieval without Relevance Labels," arXiv:2212.10496, 2022.
[7] W. Shi et al., "REPLUG: Retrieval-Augmented Black-Box Language Models," arXiv:2301.12652, 2023.
[8] A. Asai et al., "Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection," ICLR, 2024.
[9] Y. Wang et al., "Selective Knowledge Retrieval for Augmented Language Models," ACL, 2024.
[10] D. Vrandečić & M. Krötzsch, "Wikidata: A Free Collaborative Knowledge Base," Communications of the ACM, 57(10), 2014.
[11] K. Santhanam et al., "ColBERTv2: Effective and Efficient Retrieval via Lightweight Late Interaction," NAACL, 2022.
[12] T. Formal et al., "SPLADE v2: Sparse Lexical and Expansion Model for Information Retrieval," SIGIR, 2022.
[13] S. Xiao et al., "C-Pack: Packaged Resources to Advance General Chinese Embedding," arXiv:2309.07597, 2023.
[14] H. W. Chung et al., "Scaling Instruction-Finetuned Language Models," arXiv:2210.11416, 2022.
[15] P. He et al., "DeBERTa: Decoding-Enhanced BERT with Disentangled Attention," ICLR, 2021.
[16] T. Wolf et al., "Transformers: State-of-the-Art Natural Language Processing," EMNLP-Demo, 2020.
[17] N. Thakur et al., "BEIR: A Heterogeneous Benchmark for Zero-shot Evaluation of Information Retrieval Models," NeurIPS Datasets, 2021.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)