

One-Shot Translation Is Not Enough: Iterative Selection for Cross-Lingual Query Expansion

Pouya Sadeghi
University of Waterloo
Waterloo, Ontario, Canada
pouya.sadeghi@uwaterloo.ca

Jimmy Lin
University of Waterloo
Waterloo, Ontario, Canada
jimmylin@uwaterloo.ca

Abstract

We study query expansion for cross-lingual information retrieval (CLIR), where English queries are used to retrieve documents written in other languages through query translation and BM25. A translated query may still omit terms used in relevant documents. Query expansion can address this gap by using terms from retrieved documents to better match the vocabulary of the document collection. We introduce MATE, an iterative method that uses retrieved documents to generate expansion passages in the document language. For each candidate expansion, it reruns retrieval and uses large language model (LLM) relevance judgments to estimate how adding the passage changes retrieval quality, using both the original English query and its translation. Instead of selecting only the candidate with the highest estimated gain, it filters out candidates whose estimated loss in retrieval quality exceeds an allowed margin under either query. On NeuCLIR 2023–2024, MATE outperforms translation-only retrieval and xADORE, our adaptation of ADORE for expansion in the document language, across all fourteen evaluated LLM backbones. Our analysis shows that selecting the candidate with the highest estimated gain can still miss a better expansion. Although filtering may discard some useful expansions because their predicted loss exceeds the allowed margin, the average measured nDCG@20 loss among harmful candidates rejected by the translated-query check is more than five times the average gain among the other rejected candidates.

CCS Concepts

• Information systems → Query reformulation; Multilingual and cross-lingual retrieval.

Keywords

cross-lingual information retrieval, query expansion, large language models, relevance judgment, verification, reproducibility

ACM Reference Format:

Pouya Sadeghi and Jimmy Lin. 2026. One-Shot Translation Is Not Enough: Iterative Selection for Cross-Lingual Query Expansion. In *Proceedings of the 2026 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region (SIGIR-AP 2026)*, December 13–15, 2026, Singapore, Singapore. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3848000.3850209>



This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

SIGIR-AP 2026, Singapore, Singapore

© 2026 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-3033-7/2026/12

<https://doi.org/10.1145/3848000.3850209>

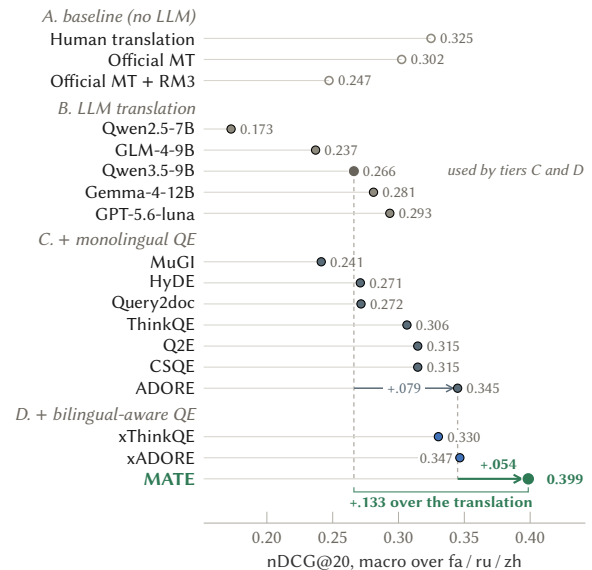


Figure 1: Translation alone is not enough. On NeuCLIR 2022 with Qwen3.5-9B, ADORE raises mean nDCG@20 from 0.266 for translation-only retrieval to 0.345, and MATE reaches 0.399, the highest score among the methods shown. Scores are averaged over Persian, Russian, and Chinese.

1 Introduction

Cross-lingual information retrieval (CLIR) retrieves documents written in a language different from that of the query. Established approaches include query translation, document translation, and language-independent representations [20, 26]. We study query translation followed by BM25 retrieval over the target-language collection. In this setting, a translated query may omit terms used in relevant documents. Query expansion before or after translation has long been used to address this problem [3]. Recent LLM-based methods provide additional ways to generate expansions [12, 34]. Figure 1 shows that query expansion can improve retrieval beyond translation alone on NeuCLIR 2022.

Iterative expansion methods use retrieved documents to guide subsequent generation. ADORE incorporates graded relevance feedback into this process [4]. Document-level assessments do not directly measure whether adding a particular expansion improves the resulting ranking. We therefore investigate whether retrieving with each candidate and assessing the resulting ranking can improve candidate acceptance.

Our initial policy selected the candidate with the largest positive predicted gain and stopped when none qualified. On NeuCLIR 2022

with GPT-5.4-mini, this policy left approximately 40% of queries unexpanded. In additional ablations with Qwen3.5-9B, continuing after a round with no accepted candidates for queries left unexpanded improved this policy’s retrieval effectiveness. We therefore examine the acceptance threshold, the number of candidates kept per round, and the effect of continuation. This investigation relates to earlier work on selective and risk-sensitive expansion [2, 6, 7].

We therefore developed MATE (Multilingual Agentic Thinking and Expansion), an iterative query expansion method for CLIR. It generates candidate expansions from retrieved documents and assesses the ranking produced by each candidate against both the translation and the original English query. The original-query assessment provides an additional check against the initial information need. We keep candidates whose estimated losses in ranking utility remain within the allowed margins under both references.

Using Qwen3.5-9B, MATE achieves mean nDCG@20 of 0.422 over NeuCLIR 2023–2024, compared with 0.385 for ADORE and 0.380 for xADORE, our adaptation of ADORE. Across all of our evaluated backbones, MATE achieves higher nDCG@20 than xADORE. Our experiments focus on BM25 as our retriever, so we compare translate-first sparse expansion methods rather than the trained cross-lingual neural retrievers studied in prior work [25, 35].

Contributions. (1) We introduce MATE, an iterative query expansion method for translate-first CLIR that assesses the rankings produced by candidate expansions against both the original query and its translation. (2) We evaluate MATE on Persian, Russian, and Chinese NeuCLIR collections across multiple LLM backbones. (3) We analyze estimated candidate utility changes and examine how acceptance thresholds, candidate caps, continuation, and the use of relevance judgments affect retrieval.

2 Related Work

CLIR and query translation. Query translation and query expansion have a long history in CLIR [3, 26]. Probabilistic structured queries incorporate alternative translations and their probabilities into estimates of term and document frequency [8]. Modern CLIR architectures combine query translation, document translation, or language-independent representations with different retrieval and reranking models [20]. We study query translation followed by target-language expansion and BM25 retrieval.

LLM query expansion. LLMs can generate expansions from the query alone [11, 12, 34, 36] or use retrieved text as context [18, 28]. Generative relevance feedback constructs feedback models from generated text [22]. ThinkQE iteratively refines expansions using corpus feedback [19], while ADORE uses graded relevance feedback [4]. Generative expansion with multilingual LLMs has also been evaluated for CLIR [23]. Deciding which expansions to use has precedents in selective and risk-sensitive query expansion. Earlier selective expansion compares original and expanded retrieval results to decide which results to return [7], while robust optimization explicitly models expansion risk [6]. MILL ranks generated and retrieved documents using agreement scores [13]. EAR trains a query reranker to predict the ranks of relevant passages under expanded queries [5]. MATE uses LLM relevance assessments against

the original and translated queries to estimate the quality of each candidate’s resulting ranking and guide candidate acceptance.

LLM relevance judgment. Previous work examines LLMs as relevance assessors [10, 31]. A large-scale study on the TREC 2024 RAG Track finds that relevance judgments generated using UMBRELA produce system rankings that correlate well with those based on human judgments [32]. We use a fixed UMBRELA-style prompt [33] and evaluate whether the resulting grades support candidate acceptance decisions.

3 MATE: Iterative Expansion and Selection

We study cross-lingual information retrieval (CLIR) of Persian, Russian, and Chinese documents using English queries. For each retrieval task, the document collection is in a target language $L \in \{\text{fa, ru, zh}\}$. Let q_s be the initial English query and q_t be its translation into L . We follow the query-translation setup described by Lin et al. [20], where we translate q_s into q_t and then retrieve documents using BM25 [27] from the target-language collection.

MATE (Multilingual Agentic Thinking and Expansion) is an iterative query expansion method for CLIR. We generate short pseudo-passages in the target language using retrieved evidence and consider each passage as a candidate expansion. We examine changes in the estimated utility of the rankings these candidates produce to decide which candidates to keep. We assess relevance against both the translated query and the original English query. Figure 2 illustrates one iteration of MATE and Figure 3 shows a worked example for a Persian retrieval query.

MATE starts by initializing the expansion memory M of accepted expansion passages and the assessment pool P of retrieved documents with their relevance grades as empty. In each round, it composes the current query from q_t and M , retrieves documents, and uses the current ranking and assessment pool to generate new candidate expansions. For each expansion, we run a script-compliance check [24] and remove candidates that fail it. We then retrieve with each remaining candidate and assess the documents from the current ranking and the candidate rankings against the original query q_s and its translation q_t to estimate each candidate’s change in ranking utility relative to the current ranking. We add newly encountered documents to the assessment pool and add candidates satisfying both acceptance checks to M . We refer to adding a candidate to M as keeping or accepting it, and to excluding it as discarding or rejecting it. After these updates, we check quality saturation on the ranking retrieved at the start of the current round and coverage saturation from the new documents encountered. We stop the process if either condition is met. Finally, we compose the query from q_t and the updated M and retrieve the final ranking.

3.1 Retrieval and Candidate Generation

Across iterations, we maintain a list of accepted expansion passages M , which is initially empty. At round t , we compose the current retrieval query s_t from the fixed translated query q_t and the expansions in M . To create this query, we join the accepted candidate passages with single spaces to obtain expansion text E . If E is empty, the retrieval query is q_t . Otherwise, we repeat q_t before appending E . The number of repetitions depends on the length of the translated

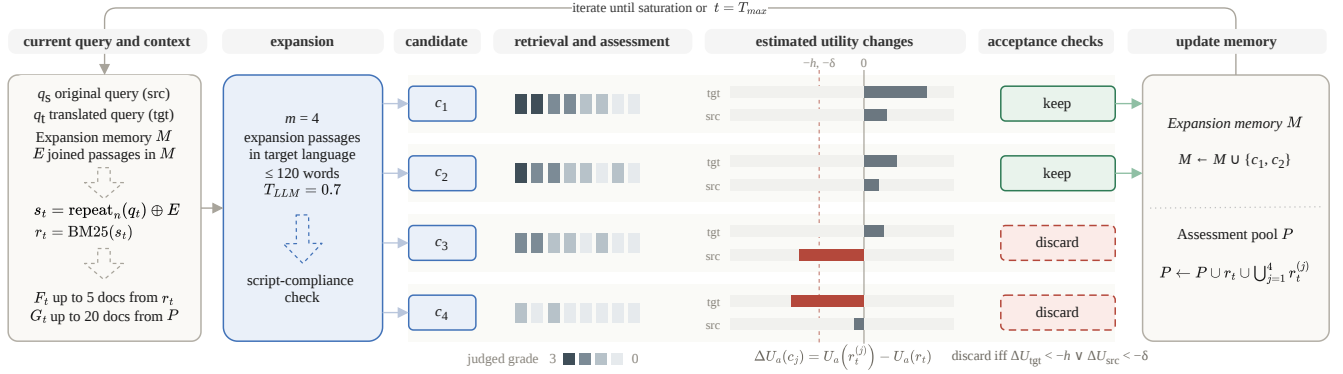


Figure 2: Candidate generation and retrieval-based evaluation in MATE. An LLM generates target-language expansions from retrieved documents, F_t , and accumulated relevance feedback, G_t . For each candidate, MATE expands the current query and reruns BM25. An LLM assesses the retrieved documents’ relevance against both reference queries. Accepted expansions and newly assessed documents are added to expansion memory M and assessment pool P for subsequent rounds.

query q_t and the expansion text E :

$$n = \max\left(1, \left\lfloor \frac{|E|}{\alpha|q_t|} \right\rfloor\right), \quad \alpha = 5.$$

We count lengths in Unicode characters. α controls the relative amount of query text. For sufficiently long expansions, the repeated translation has approximately one-fifth as many characters as E .

At each round t , we run BM25 with the query s_t and retrieve the top 20 documents. From this ranked list, we select up to five documents, excluding document identifiers selected during previous rounds. We denote this group by F_t and supply the first 1,200 characters of each selected document to the LLM backbone for generating expansions. This selection is independent of relevance assessment, so a previously assessed document remains eligible in this step if it has not been selected previously for generation.

For the example in Figure 3, q_s is “Greta Thunberg’s speech at UN”, and q_t is its Persian translation. In the first round, both the expansion memory M and assessment pool P are empty, so we use $s_1 = q_t$ for retrieval. We retrieve the top 20 documents from the Persian collection using BM25. We select up to five of these documents to form F_1 . The LLM uses their chunks to generate candidate expansion passages.

After candidate retrieval and assessment, newly encountered documents, including those retrieved by discarded candidates, are added to the pool. Each document in the assessment pool P has an LLM-estimated relevance grade from 0 to 3 against the translated query, where higher grades mean greater relevance. In each subsequent round t , we select up to 20 documents from this pool in decreasing order of target-query relevance grade and denote this group by G_t . We extract a chunk of up to 1,200, 800, 400, or 400 characters from each selected document with grade 3, 2, 1, or 0, respectively. We group these excerpts by grade and provide them to the LLM alongside the excerpts from F_t . The prompt asks the model to reuse vocabulary from documents with grades 2 and 3, and to avoid vocabulary from grades 0 and 1, and to cover material not already represented in M . Our pool selection does not reserve slots for particular grades. Graded passages can recur across rounds and may overlap with those selected from the current ranking (F_t).

For each generation round, we ask the LLM to produce candidate expansions in the target language, each approximately 120 words or fewer, and sample $m = 4$ completions at temperature 0.7. Because generated expansions may not follow the requested language [24], we apply a script check before evaluating them. A candidate passes when at least half of its alphabetic characters lie in the script associated with the target language. However, script compliance alone does not establish language identity when languages share a script.

For the Greta Thunberg query, we generate four Persian candidate passages, $c_1, \dots, c_4$, using the retrieved evidence. Figure 3 shows their English summaries. For example, c_1 describes her emotional speech, the betrayal of her generation, and the 1.5°C target. All four candidates pass the script check, so we evaluate each one.

3.2 Utility Estimation

For each candidate passage c_j that passes the script check, using the same composition rule as for s_t , we compose a query $s_t^{(j)}$ from q_t and the accepted expansion passages in M followed by c_j . Let r_t and $r_t^{(j)}$ be the top-20 rankings for the current query s_t and candidate query $s_t^{(j)}$, respectively. We assess every candidate against the same current query, before adding any candidate from that round to M .

For candidate c_1 in Figure 3, we combine its Persian passage with q_t , using the same repetition rule, to form $s_1^{(1)}$. We retrieve again with BM25 and compare the resulting top-20 ranking, $r_1^{(1)}$, with the translation-only ranking, r_1 . We evaluate the other three candidates in the same way, each against r_1 .

We assess each distinct document from the current and candidate rankings using an LLM with a fixed UMBRELA-style prompt [31, 33] at temperature 0. The assessor receives up to the first 2,000 characters of each document. For each document, we get two relevance grades from 0 to 3, one against the translated query q_t and one against the original query q_s . Documents remain in the target language in both assessments. We use the original query to provide an additional check against the original information need when selecting expansions.

For reference $a \in \{\text{tgt}, \text{src}\}$, let $g_a(d)$ be the LLM-predicted relevance grade of document d . For a retrieved ranking $r = (d_1, \dots, d_{|r|})$,

English: Greta Thunberg’s speech at UN
Persian (English gloss): Greta Thunberg’s speech at the United Nations

Retrieval and generation: We retrieve Persian documents with BM25 and use evidence from up to five of them to generate four candidate passages.

Round 1: estimated DCG@20 changes

Candidate passage (English summary)	Target	Source	Decision
c_1 . Emotional speech; betrayal of her generation, greenhouse-gas failure, 1.5 °C.	+1.56	+0.48	keep
c_2 . France 24 report mentioning environment minister Catherine McKenna.	+2.01	+0.64	keep
c_3 . Truncated near-duplicate of candidate c_1 , ending mid-sentence.	+0.07	-1.11	discard
c_4 . New York venue and a criticism framing.	+1.55	+1.78	keep

We add c_1 , c_2 , and c_4 to the expansion memory M . We discard c_3 because its estimated source-query loss exceeds 1.0.

Final nDCG@20: 0.556 for MATE (3 rounds, 11 accepted passages), 0.400 for xADORE, and 0.217 for translation-only retrieval.

Figure 3: Worked example for NeuCLIR 2022 Persian topic 126 using Qwen3.5-9B. We show English summaries of the Persian candidate passages. Target and source refer to the assessment against the translated query and the original query, respectively. Each candidate is evaluated separately against the translation-only ranking in the first round.

we estimate its utility using unnormalized DCG@20:

$$U_a(r) = \sum_{i=1}^{\min(20, |r|)} \frac{2^{g_a(d_i)} - 1}{\log_2(i+1)}. \quad (1)$$

We estimate the change associated with adding candidate c_j by comparing its ranking with the current ranking:

$$\Delta U_a(c_j) = U_a(r_t^{(j)}) - U_a(r_t).$$

We compute the difference separately for each query reference. For candidate c_1 in Figure 3, the estimated DCG@20 changes are $\Delta U_{\text{tgt}}(c_1) = 1.56$ against the Persian translation and $\Delta U_{\text{src}}(c_1) = 0.48$ against the original English query. Both changes are relative to the translation-only ranking r_1 . In this case, adding c_1 increases the estimated DCG@20 under both queries.

3.3 Candidate Acceptance and Iterations

In each iteration, we apply a harm-filter policy, under which we keep candidates whose estimated loss in ranking utility is within the allowed margin for both query references and discard the remaining candidates. We denote the allowed loss against the translated query by h and against the original query by δ . We keep a candidate when

$$\Delta U_{\text{tgt}}(c_j) \geq -h \quad \text{and} \quad \Delta U_{\text{src}}(c_j) \geq -\delta.$$

We use absolute margins $h = \delta = 1.0$ on the estimated DCG@20 scale. After assessment, we add all candidate expansions that pass both acceptance checks to M . This rule tolerates small predicted losses and does not guarantee improvements in effectiveness.

For the Greta Thunberg example in Figure 3, we keep c_1 , c_2 , and c_4 because they pass both checks, and add them to M . For candidate c_3 , the estimated DCG@20 increases by 0.07 against q_t but decreases by 1.11 against q_s . Its source-query loss exceeds the allowed margin of 1.0, so we discard it.

In the next round, we combine q_t with the three kept passages to form s_2 , using the same repetition rule. The assessment pool P

contains the documents assessed from the initial retrieval and all four candidate retrievals, including those retrieved using c_3 . We use their relevance grades against q_t to select feedback passages for the next generation round.

Continuation and termination. Iterations continue until the iteration budget is exhausted or a termination condition is met. We stop if two consecutive rounds produce no new candidate that passes our checks. The process also stops when the first ten documents in the current ranking all receive grade 3 against the translated query (quality saturation), or when two consecutive rounds add no new documents to the target assessment pool (coverage saturation). We check quality saturation after the round’s assessment and acceptance steps. Coverage counts new documents in the union of the current and candidate rankings.

We compose the final retrieval query from q_t and the updated M , using the same composition rule, and run BM25 with this query.

For the Greta Thunberg query in Figure 3, we run three rounds and accept 11 passages. The final query combines these passages with the Persian translation and achieves nDCG@20 of 0.556, compared with 0.217 for translation-only retrieval.

Computation. Each iteration performs one retrieval for the current query and up to $m = 4$ candidate retrievals. Three iterations therefore require at most 15 retrievals within the expansion process. Additional work includes one translation, up to four sampled generation completions per iteration, and relevance assessment against both query references. We reuse document assessments across iterations, so the total assessment work depends on the distinct documents encountered during the run.

4 Experimental Setup

Collections and evaluation design. We evaluate on the Persian, Russian, and Chinese news collections from TREC NeuCLIR [15–17], using English title queries. NeuCLIR 2022 provides the development data, with 46, 45, and 49 topics in Persian, Russian, and Chinese, respectively. The corresponding counts are 60/62/62 for 2023 and 68/65/56 for 2024. We report mean nDCG@20 over the 373 topic-language pairs in NeuCLIR 2023–2024.

Retrieval implementation. We use BM25 [27] with $k_1 = 0.9$, $b = 0.4$ through Pyserini [21]. We rebuild the indexes from the document collections using language-specific analyzers. For final retrieval, we return the top 1,000 documents.

LLM backbones. We evaluate fourteen LLM backbones on NeuCLIR 2022–2024, as listed in Tables 2 and 3. For MATE, we use the same model for translation, expansion generation, and relevance assessment. Translation and assessment use temperature 0, while expansion generation uses temperature 0.7. We serve the open models locally using vLLM [14].

Baselines. We compare MATE with translation-only BM25 using each model’s translation, BM25 using the official human and machine translations, and RM3 [1] applied to the machine-translated queries. We also evaluate LLM-based query expansion methods using BM25. For each method, we first translate the query and then apply query expansion to the translation.

Methods such as MuGI [36], Query2doc [34], and our BM25 adaptation of HyDE [11] generate pseudo-documents from the translated query without any corpus interaction. Corpus-conditioned methods, such as CSQE [18], first retrieve documents with the translated query and use them to guide expansion. Similarly, iterative methods, such as ThinkQE [19] and ADORE [4], start from the translated query and refine the expansions using corpus feedback.

To examine the effect of an explicit target-language instruction separately from MATE’s candidate assessment and selection, we introduce xThinkQE and xADORE, our adaptations of ThinkQE and ADORE, respectively. Both original baselines receive the translated query; our adaptations additionally instruct the LLM to generate expansion passages in the document language L . We use the Expandorini implementations of these query expansion baselines and the same LLM for translation and expansion in each configuration.

Tables 1 and 3 report comparisons on NeuCLIR 2023–2024. In Tables 2 and 3, Δ is MATE’s score minus the higher score of xThinkQE and xADORE for each backbone.

Alternative acceptance policies. We compare the acceptance policy in §3.3 with three alternatives. A *capped* version keeps at most a fixed number of candidates per round. Among candidates satisfying both acceptance conditions, it keeps those with the highest estimated target-query utility gains until the cap is reached. The *select-best* policy chooses the candidate with the largest positive target-query gain among those passing the source-query check. The *keep-improving* policy keeps every candidate with a strictly positive estimated target-query gain that also passes the source-query check. Table 5 reports the policy results.

Additionally, we evaluate relative margins that scale the allowed losses with the current ranking’s estimated utility. The target- and source-query margins are

$$h_{\text{rel}} = \max\{0.15 U_{\text{tgt}}(r_t), 0.5\},$$

$$\delta_{\text{rel}} = \max\{0.25 U_{\text{src}}(r_t), 0.5\}.$$

These replace h and δ in the acceptance checks.

Verification controls. We compare three variants of how relevance judgments are used. In *accept-all*, we keep every candidate that passes the script check, regardless of its estimated utility. Relevance grades later guide candidate generation and stopping, but they do not affect which candidates are kept or discarded.

The *no-verify* control is similar, but skips retrieval for individual candidates and relevance assessment steps. Without relevance grades, utility-based stopping is also unavailable, so we stop at the round limit or when no candidate passes the script check.

In *random-keep*, we retrieve and assess candidates, but independently keep each candidate passing the script check with probability 0.9. Relevance grades later guide candidate generation and stopping, but do not affect which candidates are kept or discarded.

Metrics and statistical analysis. We use nDCG@20 as our primary effectiveness metric and also report recall at 1,000. Relative to translation-only retrieval, we classify a query as harmed when its nDCG@20 decreases by more than 0.05 and improved when it increases by more than 0.05. We compare systems using paired randomization tests on per-query nDCG@20 differences [29], with 10^5 permutations, a fixed seed, and an add-one correction.

Table 1: nDCG@20 on NeuCLIR 2023–2024 using Qwen3.5-9B and BM25. Overall reports mean nDCG@20 over all topic–language pairs from both years. Bold marks the highest score in each column.

Method	NeuCLIR 2023			NeuCLIR 2024			Overall
	fa	ru	zh	fa	ru	zh	
Translation only	.374	.355	.166	.355	.266	.164	.282
Query2doc	.386	.396	.175	.359	.355	.209	.316
HyDE	.351	.378	.173	.320	.338	.174	.291
MuGI	.312	.359	.191	.294	.301	.218	.280
Q2E	.361	.394	.208	.334	.358	.230	.316
CSQE	.410	.425	.220	.416	.388	.255	.355
ThinkQE	.391	.394	.235	.367	.373	.244	.336
xThinkQE	.414	.449	.225	.435	.417	.233	.365
ADORE	.401	.477	.238	.455	.431	.294	.385
xADORE	.451	.483	.227	.408	.418	.283	.380
MATE	.458	.467	.291	.506	.465	.328	.422

Judge validation. We evaluate Qwen2.5-7B-Instruct [30] and Qwen3.5-9B as relevance assessors on 210 topic–document pairs per language from NeuCLIR 2022, using both the original English query and its human translation. We sample 70 pairs from each human relevance grade. Against human relevance judgments, Spearman correlation ranges from 0.46 to 0.72, binary F1 from 0.79 to 0.92, precision from 0.88 to 0.96, and recall from 0.70 to 0.90. This evaluation measures agreement on document relevance; §6.1 examines the estimated utility changes used for candidate acceptance.

5 Results

5.1 Overall Effectiveness

Table 1 compares query expansion methods on NeuCLIR 2023 and 2024 using Qwen3.5-9B as the backbone and BM25 as the retriever. MATE achieves mean nDCG@20 of 0.422 over both years, compared with 0.385 for ADORE, 0.380 for xADORE, and 0.365 for xThinkQE. MATE achieves the highest score among the compared methods in five of the six language–year pairs. The exception is Russian in 2023, where xADORE scores 0.483 and MATE scores 0.467. By averaging over the evaluated topic–language pairs within each year, MATE outperforms xADORE by +0.019 in 2023 and +0.065 in 2024.

5.2 Effect of Model Backbones

Table 2 reports NeuCLIR 2022 results across fourteen backbones, using macro-averaged nDCG@20 over the three languages. MATE scores above the strongest of xThinkQE and xADORE on twelve backbones. The exceptions are Qwen2.5-3B and GLM-4-9B, with differences of -0.020 and -0.031 , respectively. Table 3 reports the NeuCLIR 2023 and 2024 results for fourteen backbones. MATE achieves higher mean nDCG@20 over NeuCLIR 2023–2024 than xADORE on all backbones. However, for GLM-4-9B, xThinkQE scores higher than MATE on both NeuCLIR 2022 and NeuCLIR 2023–2024 evaluations; MATE nevertheless remains above xADORE. Figure 1 further shows that, with Qwen3.5-9B, MATE outperforms a broader set of query expansion baselines on NeuCLIR 2022.

Table 2: Mean nDCG@20 on NeuCLIR 2022. *Transl.* is each model’s translation. Bold marks the highest reported score in each row. Δ is the margin over the row’s best baseline.

Backbone	Transl.	xThinkQE	xADORE	MATE	Δ
<i>Qwen2.5</i>					
1.5B	.093	.118	.102	.120	+0.02
3B	.103	.132	.160	.140	−.020
7B	.173	.237	.222	.250	+0.013
14B	.191	.255	.312	.341	+0.029
32B	.227	.288	.333	.374	+0.041
72B	.257	.310	.349	.397	+0.048
<i>Gemma-4</i>					
E4B	.281	.327	.344	.377	+0.033
12B	.281	.338	.372	.408	+0.036
31B	.311	.366	.413	.425	+0.012
<i>Qwen3.x</i>					
3.5-9B	.266	.330	.347	.399	+0.052
3.6-35B-A3B	.271	.336	.390	.417	+0.027
<i>GLM-4</i>					
9B	.237	.283	.244	.252	−.031
<i>Closed frontier</i>					
GPT-5.4-mini	.294	.354	.396	.412	+0.016
GPT-5.6-luna	.293	.366	.394	.420	+0.026

Table 3: Mean nDCG@20 on NeuCLIR 2023–2024. *Transl.* is each model’s translation. Bold marks the highest reported score in each row. Δ is the margin over the row’s best baseline.

Backbone	Transl.	xThinkQE	xADORE	MATE	Δ
<i>Qwen2.5</i>					
1.5B	.093	.115	.102	.116	+0.001
3B	.131	.164	.180	.182	+0.002
7B	.178	.250	.273	.279	+0.006
14B	.212	.298	.343	.366	+0.023
32B	.230	.325	.352	.399	+0.047
72B	.271	.350	.396	.429	+0.033
<i>Gemma-4</i>					
E4B	.289	.359	.369	.387	+0.018
12B	.302	.381	.392	.443	+0.051
31B	.317	.398	.436	.471	+0.035
<i>Qwen3.x</i>					
3.5-9B	.282	.365	.380	.422	+0.042
3.6-35B-A3B	.305	.387	.421	.435	+0.014
<i>GLM-4</i>					
9B	.246	.282	.256	.265	−.017
<i>Closed frontier</i>					
GPT-5.4-mini	.320	.423	.438	.466	+0.028
GPT-5.6-luna	.327	.429	.425	.474	+0.045

5.3 Effect of Target-Language Adaptations

We compare ThinkQE and ADORE with our target-language adaptations, xThinkQE and xADORE. Using Qwen3.5-9B as the backbone over NeuCLIR 2023–2024, xThinkQE achieves mean nDCG@20 of

Table 4: Mean target-script character ratios (%) for Qwen3.5-9B on NeuCLIR 2023–2024. Each value averages the per-passage fraction of alphabetic characters in the target script over all expansion passages generated across iterations.

Method	Persian	Russian	Chinese
ThinkQE	84.23	69.64	77.76
xThinkQE	98.05	97.96	93.25
ADORE	93.77	87.67	81.15
xADORE	98.10	96.66	92.98

	Measured retrieval change			
	Harmful		Not harmful	
Fails the check	<i>n</i> =59	−0.193	<i>n</i> =130	+0.036
Passes the check	<i>n</i> =38	−0.207	<i>n</i> =333	+0.100

Figure 4: Target-query check for the first-round candidates on NeuCLIR 2022 (Qwen3.5-9B; 560 candidates, 140 queries). A candidate fails the check when its estimated DCG@20 change is below −1.0 and is classified as harmful when its measured nDCG@20 change is below −0.05. Cells show candidate counts and mean measured changes in nDCG@20 relative to the translation-only ranking.

0.365, compared with 0.336 for ThinkQE, and xADORE achieves 0.380, compared with 0.385 for ADORE (Table 1).

To further investigate how this instruction affects the generated text, we analyze all 20,803 expansion passages generated across iterations in these runs. Table 4 shows that both adaptations increase the mean proportion of alphabetic characters in the target script in all three languages. These results suggest that an explicit target-language instruction improves script consistency, but its benefit for retrieval depends on the expansion method.

6 Analysis

We examine how well the estimated utility changes identify harmful expansions, which components affect retrieval, and how the acceptance policies perform across backbones. For all of our analysis, unless stated otherwise, we use NeuCLIR 2022.

6.1 Candidate Utility Estimation

Figure 4 compares estimated target-query utility changes with measured changes in nDCG@20 for 560 first-round candidates from 140 queries using Qwen3.5-9B as the backbone. We evaluate each candidate against the same translation-only ranking for its query. A candidate is harmful when its measured nDCG@20 change is below −0.05. The target-query check rejects candidates whose estimated DCG@20 loss exceeds 1.0. This analysis considers only the target-query check and not the final acceptance decision.

In our case study, the target-query check rejects 189 candidates, of which 59 meet our harm criterion, so the check has a precision of 0.31 and recall of 0.61. This means that this check rejects 130 candidates that do not meet our harm criterion and can actually be helpful for the method. However, numbers show that this is a trade-off worth making; the harmful rejected candidates reduce

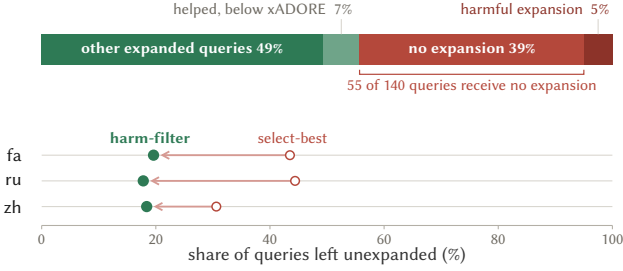


Figure 5: The upper panel shows select-best outcomes on NeuCLIR 2022. Helped queries outperform translation-only retrieval but remain below xADORE. The lower panel compares proportions of unexpanded queries under the select-best and harm-filter policies. Both panels use GPT-5.4-mini.

nDCG@20 by 0.193 on average, while the other rejected candidates improve it by 0.036 on average. The average loss in the first group is about 5.4 times the average gain in the second. Considering all rejected candidates together, their average change in nDCG@20 is -0.036 , while candidates that pass the check have an average gain of 0.068. The check also reduces the proportion of harmful candidates from 17.3% in the original set to 10.2% among those that pass. So, although we reject some helpful expansions, the remaining candidates have higher average gains and are less often harmful.

We also check whether the candidate with the highest estimated gain actually produces the best retrieval result. For each query, we expand the translated query with each of the four candidates generated in the first round and measure nDCG@20 for each resulting ranking. We then check whether the candidate with the highest estimated gain also achieves the highest measured nDCG@20. This holds for 73 of 140 queries (52%). Across all 140 queries, this candidate’s nDCG@20 is on average 0.026 below the highest score among the four candidates. These results show that selecting the candidate with the highest estimated gain can miss a better expansion.

6.2 Component Ablations

Table 5 compares acceptance policies, verification controls, and iteration budgets using Qwen3.5-9B as the backbone. The last block modifies the stopping harm-filter variant instead of ablating MATE’s configurations.

Verification controls. MATE achieves macro nDCG@20 of 0.399, which drops to 0.335 without verification, 0.361 with the accept-all policy, and 0.374 with random-keep. MATE scores above all three controls in each language, even though random-keep accepts more passages than MATE (10.2 versus 8.0 per query) and has higher recall at 1,000 (0.689 versus 0.681). This suggests that adding more expansion alone does not account for the improvement. A similar pattern holds when we use GPT-5.4-mini; MATE scores above random-keep in all three languages. The macro difference is approximately $+0.025$ for both backbones, with $p = 0.11$ for Qwen3.5-9B and $p = 0.016$ for GPT-5.4-mini. These comparisons support using the estimated utility changes for acceptance.

Termination and continuation. Another design choice is when to stop expanding a query. Stopping too early can leave queries unexpanded and underuse the available budget, while continuing

expansion can consume additional computation without improving retrieval quality. We examine this issue for the select-best policy, where we terminate the process once no candidate meets the acceptance criteria. With GPT-5.4-mini, this policy leaves 55 of 140 queries (39%) without an expansion. On the other hand, when we use the same termination rule with the harm-filter policy, it leaves fewer queries unexpanded in all three languages (Figure 5). This shows that the acceptance policy also affects whether a query receives any expansion. We therefore examine what happens when the loop continues after a round with no accepted candidates.

Using Qwen3.5-9B as the backbone, enabling continuation raises select-best from 0.345 to 0.401 macro nDCG@20. The change for the harm-filter is from 0.368 to 0.399. Therefore, for our termination policy, a round with no accepted candidates does not terminate the process by itself. Instead, the document assessments collected so far are utilized to guide candidate generation in subsequent rounds. We terminate the process when the iteration budget is exhausted, two consecutive rounds produce no new accepted candidate, or either quality or coverage saturation is reached.

6.3 Acceptance Policies Across Backbones

We also investigate whether the same acceptance policy performs best across different backbones. With Qwen3.5-9B and GPT-5.4-mini, MATE scores above random-keep in all three languages. However, using Qwen2.5-7B-instruct as our backbone, random-keep achieves nDCG@20 scores of 0.168/0.385/0.209 for Persian/Russian/Chinese, compared with 0.153/0.380/0.216 for MATE. On NeuCLIR 2022, the mean nDCG@20 difference between MATE and random-keep is -0.004 and is not statistically significant ($p = 0.77$). Keep-improving achieves 0.177/0.405/0.222, scoring above both policies in all three languages. We attribute this difference to the quality of the relevance judgments, although the backbone variation affects the translations and generated expansions as well.

7 Limitations

Our experiments focus on English queries and Persian, Russian, and Chinese news collections, using query translation followed by BM25 retrieval. We have not evaluated MATE with dense retrievers, hybrid retrieval, or reranking systems, including those used in NeuCLIR [25, 35]. Moreover, our reported results mostly use one generation run per configuration. Additional runs would help assess variation across generation seeds [9]. The acceptance policies perform differently across backbones. To support our findings, we need more reliable estimates of improvement based on results averaged across multiple seeds. Also, differences in judge quality may help explain these results, but changing the backbone also changes translation and expansion generation. Further experiments can help separate these effects.

8 Conclusion

Query translation provides a query in the document language, but its terms may still differ from those used in relevant documents. We introduced MATE, an iterative query expansion method that uses retrieved documents to generate target-language expansions and evaluates each candidate using the ranking it produces. We assess relevance against both the translated query and the original

Table 5: Acceptance policies and ablations on NeuCLIR 2022 using Qwen3.5-9B (one generation run). The fa, ru, zh, and Macro columns report nDCG@20. Macro and R@1k average over languages. Avg. passages is the mean expansion-passage count per query, and Unexpanded (%) is the percentage of queries without expansion. Δ Macro is computed from the displayed Macro scores relative to the base named in each block. MATE uses up to three iterations with four candidates per iteration.

Policy	Target boundary	Cap	Continue	fa	ru	zh	Macro	R@1k	Avg. passages	Unexpanded queries (%)	Δ Macro
MATE	$\geq -h$	∞	Yes	.436	.464	.296	.399	.681	8.0	0.0	—
<i>xADORE</i>	(for reference)			.373	.399	.268	.347	.667	5.0	0.0	-.052
<i>Translation only</i>	(for reference)			.314	.338	.148	.266	.596	0.0	100.0	-.133
1. Acceptance policy (base: MATE)											
select-best	> 0	1	No	.396	.400	.239	.345	.639	1.2	40.7	-.054
select-best	> 0	1	Yes	.433	.464	.307	.401	.693	2.0	0.0	+.002
keep-improving	> 0	∞	Yes	.415	.451	.316	.394	.685	5.9	0.0	-.005
harm-filter	$\geq -h$	∞	No	.421	.409	.276	.368	.667	7.1	27.1	-.031
harm-filter	$\geq -h$	1	Yes	.436	.426	.292	.385	.686	2.4	0.0	-.014
MATE + relative margins	$\geq -h_{rel}$	∞	Yes	.440	.456	.282	.393	.680	10.1	0.0	-.006
2. Verification controls (base: MATE)											
accept-all	—	∞	—	.390	.427	.265	.361	.676	11.4	0.0	-.038
no-verify	(no relevance judgments)			.385	.411	.209	.335	.659	12.0	0.0	-.064
random-keep	(keep probability .9)			.420	.430	.271	.374	.689	10.2	0.0	-.025
3. Iteration (base: MATE)											
rounds = 1	$\geq -h$	∞	—	.373	.405	.190	.323	.655	2.7	0.0	-.076
4. Other settings (base: stopping harm-filter)											
candidates $m = 2$	$\geq -h$	∞	No	.421	.406	.243	.357	.667	3.4	29.3	-.011
harm margin $h = .5$	$\geq -.5$	∞	No	.413	.404	.263	.360	.671	6.5	28.6	-.008
harm margin $h = 2.0$	≥ -2.0	∞	No	.408	.431	.272	.371	.684	8.0	16.4	+.003
no source-query check	$\geq -h$	∞	No	.410	.435	.244	.363	.673	8.1	16.4	-.005

English query, and keep candidates whose estimated losses remain within the allowed margins. Using Qwen3.5-9B as the backbone, MATE achieves mean nDCG@20 of 0.422 over NeuCLIR 2023–2024, compared with 0.385 for ADORE and 0.380 for xADORE.

During our experiments, we found that the estimated gains provide useful information, but do not consistently identify the best candidate. We also found that stopping after a round with no accepted candidates can leave many queries unexpanded. Adding continuation improves both select-best and the harm-filter in our ablations. These results show that checking candidates is useful, but how we accept expansions and decide when to stop also matters for retrieval effectiveness.

Acknowledgments

This research was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC). Additional funding was provided by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean Government (MSIT) (No. RS-2024-00457882, National AI Research Lab Project).

References

- [1] Nasreen Abdul-Jaleel, James Allan, W. Bruce Croft, Fernando Diaz, Leah S. Larkey, Xiaoyan Li, Mark D. Smucker, and Courtney Wade. 2004. UMass at TREC 2004: Novelty and HARD. In *Proceedings of the Thirteenth Text REtrieval Conference, TREC 2004, Gaithersburg, Maryland, USA, November 16-19, 2004 (NIST Special Publication, Vol. 500-261)*, Ellen M. Voorhees and Lori P. Buckland (Eds.). National Institute of Standards and Technology (NIST). <http://trec.nist.gov/pubs/trec13/papers/umass.novelty.hard.pdf>
- [2] Giambattista Amati, Claudio Carpineto, and Giovanni Romano. 2004. Query Difficulty, Robustness, and Selective Application of Query Expansion. In *Advances in Information Retrieval*, Sharon McDonald and John Tait (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 127–137.
- [3] Lisa Ballesteros and W. Bruce Croft. 1997. Phrasal Translation and Query Expansion Techniques for Cross-Language Information Retrieval. In *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Philadelphia, Pennsylvania, USA) (SIGIR '97). Association for Computing Machinery, New York, NY, USA, 84–91. doi:10.1145/258525.258540
- [4] Amin Bigdeli, Negar Arabzadeh, Radin Hamidi Rad, Sajad Ebrahimi, Charles L. A. Clarke, and Ebrahim Bagheri. 2026. ADORE: Iterative Query Expansion with Retrieval-Grounded Relevance Feedback. arXiv:2606.13905 [cs.LR] <https://arxiv.org/abs/2606.13905>
- [5] Yung-Sung Chuang, Wei Fang, Shang-Wen Li, Wen-tau Yih, and James Glass. 2023. Expand, Rerank, and Retrieve: Query Reranking for Open-Domain Question Answering. 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, 12131–12147. doi:10.18653/v1/2023.findings-acl.768
- [6] Kevyn Collins-Thompson. 2009. Reducing the Risk of Query Expansion via Robust Constrained Optimization. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management (Hong Kong, China) (CIKM '09)*. Association for Computing Machinery, New York, NY, USA, 837–846. doi:10.1145/1645953.1646059
- [7] Steve Cronen-Townsend, Yun Zhou, and W. Bruce Croft. 2004. A Framework for Selective Query Expansion. In *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management* (Washington DC, USA) (CIKM '04). Association for Computing Machinery, New York, NY, USA, 236–237. doi:10.1145/1031171.1031220
- [8] Kareem Darwish and Douglas W. Oard. 2003. Probabilistic Structured Query Methods. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Toronto, Canada) (SIGIR

- '03). Association for Computing Machinery, New York, NY, USA, 338–344. doi:10.1145/860435.860497
- [9] Jesse Dodge, Suchin Gururangan, Dallas Card, Roy Schwartz, and Noah A. Smith. 2019. Show Your Work: Improved Reporting of Experimental Results. 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)*, Kentaro Iyui, Jing Jiang, Vincent Ng, and Xiaojun Wan (Eds.). Association for Computational Linguistics, Hong Kong, China, 2185–2194. doi:10.18653/v1/D19-1224
- [10] Guglielmo Faggioli, Laura Dietz, Charles L. A. Clarke, Gianluca Demartini, Matthias Hagen, Claudia Hauff, Noriko Kando, Evangelos Kanoulas, Martin Potthast, Benno Stein, and Henning Wachsmuth. 2023. Perspectives on Large Language Models for Relevance Judgment. In *Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval (Taipei, Taiwan) (ICTIR '23)*. Association for Computing Machinery, New York, NY, USA, 39–50. doi:10.1145/3578337.3605136
- [11] 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. doi:10.18653/v1/2023.acl-long.99
- [12] Rolf Jagerman, Honglei Zhuang, Zhen Qin, Xuanhui Wang, and Michael Bendersky. 2023. Query Expansion by Prompting Large Language Models. arXiv:2305.03653 [cs.IR] <https://arxiv.org/abs/2305.03653>
- [13] Pengyue Jia, Yiding Liu, Xiangyu Zhao, Xiaopeng Li, Changying Hao, Shuaiqiang Wang, and Dawei Yin. 2024. MLL: Mutual Verification with Large Language Models for Zero-Shot Query Expansion. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, Kevin Duh, Helena Gomez, and Steven Bethard (Eds.). Association for Computational Linguistics, Mexico City, Mexico, 2498–2518. doi:10.18653/v1/2024.naacl-long.138
- [14] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the 29th ACM Symposium on Operating Systems Principles (SOSP 2023)*. 611–626. doi:10.1145/3600066.3613165
- [15] Dawn Lawrie, Sean MacAvaney, James Mayfield, Paul McNamee, Douglas W. Oard, Luca Soldaini, and Eugene Yang. 2022. Overview of the TREC 2022 NeuCLIR Track. In *Proceedings of the Thirty-First Text REtrieval Conference (TREC 2022)*.
- [16] Dawn Lawrie, Sean MacAvaney, James Mayfield, Paul McNamee, Douglas W. Oard, Luca Soldaini, and Eugene Yang. 2023. Overview of the TREC 2023 NeuCLIR Track. In *Proceedings of the Thirty-Second Text REtrieval Conference (TREC 2023)*.
- [17] Dawn Lawrie, Sean MacAvaney, James Mayfield, Paul McNamee, Douglas W. Oard, Luca Soldaini, and Eugene Yang. 2024. Overview of the TREC 2024 NeuCLIR Track. In *Proceedings of the Thirty-Third Text REtrieval Conference (TREC 2024)*.
- [18] Yibin Lei, Yu Cao, Tianyi Zhou, Tao Shen, and Andrew Yates. 2024. Corpus-Steered Query Expansion with Large Language Models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, Yvette Graham and Matthew Purver (Eds.). Association for Computational Linguistics, St. Julian's, Malta, 393–401. doi:10.18653/v1/2024.eacl-short.34
- [19] Yibin Lei, Tao Shen, and Andrew Yates. 2025. ThinkQE: Query Expansion via an Evolving Thinking Process. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (Eds.). Association for Computational Linguistics, Suzhou, China, 17772–17781. doi:10.18653/v1/2025.findings-emnlp.965
- [20] Jimmy Lin, David Alfonso-Hermelo, Vitor Jeronymo, Ehsan Kamalloo, Carlos Lassance, Rodrigo Nogueira, Odunayo Ogundepo, Mehdi Rezagholizadeh, Nandan Thakur, Jheng-Hong Yang, and Xinyu Zhang. 2023. Simple Yet Effective Neural Ranking and Reranking Baselines for Cross-Lingual Information Retrieval. arXiv:2304.01019 [cs.IR] <https://arxiv.org/abs/2304.01019>
- [21] Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: A Python Toolkit for Reproducible Information Retrieval Research with Sparse and Dense Representations. 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, 2356–2362. doi:10.1145/3404835.3463238
- [22] Iain Mackie, Shubham Chatterjee, and Jeffrey Dalton. 2023. Generative Relevance Feedback with Large Language Models. 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, 2026–2031. doi:10.1145/3539618.3591992
- [23] Olivia Macmillan-Scott, Roksana Goworek, and Eda B. Özyiğit. 2025. Generative Query Expansion with Multilingual LLMs for Cross-Lingual Information Retrieval. arXiv:2511.19325 [cs.IR] <https://arxiv.org/abs/2511.19325>
- [24] Kelly Marchisio, Wei-Yin Ko, Alexandre Berard, Théo Dehaze, and Sebastian Ruder. 2024. Understanding and Mitigating Language Confusion in LLMs. 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, 6653–6677. doi:10.18653/v1/2024.emnlp-main.380
- [25] Suraj Nair, Eugene Yang, Dawn Lawrie, Kevin Duh, Paul McNamee, Kenton Murray, James Mayfield, and Douglas W. Oard. 2022. Transfer Learning Approaches for Building Cross-Language Dense Retrieval Models. In *Advances in Information Retrieval: 44th European Conference on IR Research, ECIR 2022, Stavanger, Norway, April 10–14, 2022, Proceedings, Part I*. Springer-Verlag, Berlin, Heidelberg, 382–396. doi:10.1007/978-3-030-99736-6_26
- [26] Jian-Yun Nie. 2010. *Cross-Language Information Retrieval*. Morgan & Claypool. doi:10.2200/S00266ED1V01Y201005HLT008
- [27] Stephen Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval* 3, 4 (April 2009), 333–389. doi:10.1561/15000000019
- [28] Tao Shen, Guodong Long, Xiubo Geng, Chongyang Tao, Yibin Lei, Tianyi Zhou, Michael Blumenstein, and Daxin Jiang. 2024. Retrieval-Augmented Retrieval: Large Language Models are Strong Zero-Shot Retriever. In *Findings of the Association for Computational Linguistics: ACL 2024*, Lun-Wei Ku, Andre Martins, and Vivek Srikumar (Eds.). Association for Computational Linguistics, Bangkok, Thailand, 15933–15946. doi:10.18653/v1/2024.findings-acl.943
- [29] Mark D. Smucker, James Allan, and Ben Carterette. 2007. A Comparison of Statistical Significance Tests for Information Retrieval Evaluation. In *Proceedings of the Sixteenth ACM Conference on Information and Knowledge Management (Lisbon, Portugal) (CIKM '07)*. Association for Computing Machinery, New York, NY, USA, 623–632. doi:10.1145/1321440.1321528
- [30] Qwen Team, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2025. Qwen2.5 Technical Report. arXiv:2412.15115 [cs.CL] <https://arxiv.org/abs/2412.15115>
- [31] Paul Thomas, Seth Spielman, Nick Craswell, and Bhaskar Mitra. 2024. Large Language Models can Accurately Predict Searcher Preferences. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (Washington DC, USA) (SIGIR '24)*. Association for Computing Machinery, New York, NY, USA, 1930–1940. doi:10.1145/3626772.3657707
- [32] Shivani Upadhyay, Ronak Pradeep, Nandan Thakur, Daniel Campos, Nick Craswell, Ian Soboroff, and Jimmy Lin. 2025. A Large-Scale Study of Relevance Assessments with Large Language Models Using UMBRELA. In *Proceedings of the 2025 International ACM SIGIR Conference on Innovative Concepts and Theories in Information Retrieval (Padua, Italy) (ICTIR '25)*. Association for Computing Machinery, New York, NY, USA, 358–368. doi:10.1145/3731120.3744605
- [33] Shivani Upadhyay, Ronak Pradeep, Nandan Thakur, Nick Craswell, and Jimmy Lin. 2024. UMBRELA: Umbrella is the (Open-Source Reproduction of the) Bing RElevance Assessor. arXiv:2406.06519 [cs.IR] <https://arxiv.org/abs/2406.06519>
- [34] Liang Wang, Nan Yang, and Furu Wei. 2023. Query2doc: Query Expansion with Large Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, Singapore, 9414–9423. doi:10.18653/v1/2023.emnlp-main.585
- [35] Eugene Yang, Dawn Lawrie, James Mayfield, Douglas W. Oard, and Scott Miller. 2024. Translate-Distill: Learning Cross-Language Dense Retrieval by Translation and Distillation. In *Advances in Information Retrieval: 46th European Conference on Information Retrieval, ECIR 2024, Glasgow, UK, March 24–28, 2024, Proceedings, Part II*. Springer-Verlag, Berlin, Heidelberg, 50–65. doi:10.1007/978-3-031-56060-6_4
- [36] Le Zhang, Yihong Wu, Qian Yang, and Jian-Yun Nie. 2024. Exploring the Best Practices of Query Expansion with Large Language Models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (Eds.). Association for Computational Linguistics, Miami, Florida, USA, 1872–1883. doi:10.18653/v1/2024.findings-emnlp.103