IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Ingénierie des Systèmes d'Information
Vol. 31, No. 5, May, 2026, pp. 1525-1533
Journal homepage: http://iieta.org/journals/isi

# Knowledge-Enriched SBERT Re-Ranking for Dialectal French Word Sense Disambiguation

Btissam El Janati[1*], Adil Enaanai[2], Fadoua Ghanimi[1], Ilyas Ghanimi[1]

[1] Department of Physics Engineering, Sustainability and Innovation Laboratory, Ibn Tofail University, Kenitra 14000, Morocco
[2] Department of Computer Science, Abdelmalek Essaadi University, Tétouan 93030, Morocco

Corresponding Author Email: Btissam.eljanati@uit.ac.ma



https://doi.org/10.18280/isi.310511

**Received:** 13 January 2026
**Revised:** 20 March 2026
**Accepted:** 20 April 2026
**Available online:** 31 May 2026

***Keywords:***
*dialectal French, word sense disambiguation, semantic search, SBERT, neural re-ranking, knowledge base, vector retrieval, zero-shot generalization*

**ABSTRACT**

French dialectal variation poses a persistent challenge for word sense disambiguation and semantic search, particularly when region-specific meanings are poorly represented in general-purpose language models. This study proposes a knowledge-enriched semantic search framework for dialectal French, combining SBERT sentence embeddings with a lightweight artificial neural network re-ranker and a curated dialectal knowledge base. The knowledge base contains 184 terms from Swiss, Belgian, and Quebec French, each linked to its dialectal meaning, standard French equivalent, and contextual enrichment terms. User queries are first corrected and expanded with dialect-specific context, then encoded with all-MiniLM-L6-v2 and retrieved through an Elasticsearch HNSW vector index. Candidate documents are re-ranked by a compact neural model that jointly uses semantic similarity and dialect match as relevance signals. The system was evaluated on 787 queries across exact-word, semantic, misspelled, long-context, and noise categories, using 600 dialect-annotated documents and comparisons with 13 retrieval and language-model baselines. The proposed framework achieved 85.9% Precision@10, an F1-score of 0.859, NDCG@10 of 0.745, and an average CPU response time of 132.5 ms. Zero-shot testing on 23 unseen dialectal terms reached 82.4% Precision@10. The results indicate that compact, knowledge-guided neural retrieval can outperform much larger general-purpose models for dialect-sensitive French disambiguation while preserving interpretability and CPU-only deplorability.

## 1. INTRODUCTION

The French language presents significant challenges for natural language processing (NLP) due to its lexical diversity, complex polysemy, and dialectal variations across regions (Swiss, Belgian, Quebecois), where the same term can carry distinct meanings (e.g., cornet means "ice cream cone" in standard French but "plastic bag" in Swiss French) [1]. Traditional search engines like Elasticsearch rely on exact word matching and statistical models, failing to capture semantic nuances [2]. While Word Sense Disambiguation (WSD) systems have been developed for standard French, they perform poorly on dialects due to limited annotated corpora and underutilization of structured resources like Wikidata and OntoLex-Lemon [3-5].

Recent large language models achieve moderate results on standard French but treat dialects implicitly, require expensive GPU infrastructure, and operate as black boxes lacking interpretability [6, 7]. Hybrid architectures combining dense and sparse representations have shown promise, including SPLADE [8], ColBERTv2 [9], and DPR [10]. However, none of these models have been specifically adapted to French dialectal sensitivity. Quantitatively, existing systems achieve at most 70.8% precision on dialectal tasks, leaving a significant performance gap to be addressed [11].

Despite advances in semantic search [2, 8-10], no existing approach simultaneously addresses four critical challenges for French dialectal NLP: (1) scarcity of dialectal annotated resources [1]; (2) opacity and computational cost of LLMs [12, 13]; (3) lack of robustness evaluation under noisy conditions; and (4) absence of interpretability in retrieval decisions [6, 7]. Furthermore, the impact of orthographic and phonetic corrections on dialectal search performance remains unexplored. This gap is measurable: current systems achieve at best 70.9% precision (Mistral Large 2), leaving substantial room for improvement.

To address these gaps, this work proposes a hybrid WSD approach that integrates contextual embeddings from SBERT (all-MiniLM-L6-v2, 22M parameters) with a lightweight Artificial Neural Network (ANN) and a dialect-aware vector search engine, enriched by a structured knowledge base of 184 dialectal terms across three dialects (Swiss CH, Belgian BE, and Quebecois QC). Unlike linear scoring functions, our ANN learns non-linear interactions between semantic similarity and dialect matching through three hidden layers (16-8-4 neurons) with ReLU activation and dropout regularization.

Our main contributions are as follows. First, we introduce a novel ANN-based architecture for dialectal French disambiguation with full interpretability through explicit query enrichment (e.g., "cornet" → "sac plastique"). Second, we provide a comprehensive evaluation against 13 state-of-

the-art models (SPLADE [8], ColBERTv2 [9], DPR [10], Mistral Large 2 [13], Gemma 3 [12], E5-multilingual [11], ModernBERT [14], GritLM [15], Jina Embeddings v3 [16], plus Elasticsearch baselines) on 787 queries and 600 documents. Third, we present the first zero-shot generalization analysis 20% of words, achieving 82.4% precision. Fourth, we conduct a robustness-to-noise analysis for dialectal French, achieving 86.8% precision on spelling error queries. Fifth, we perform an ablation study comparing lightweight query-side correction (+0.6% F1) versus heavy index-side normalization (-23.4% F1). Finally, we demonstrate CPU-only deployment with no GPU requirement, achieving a 132.5 ms response time and 65W power consumption, while outperforming Mistral Large 2 (123B parameters) by +15.0 percentage points despite being 5,600× smaller.

The remainder of this paper is organized as follows: Section I introduces the problem and motivation, Section II reviews related work, Section III describes the proposed method, Section IV presents and discusses the experimental results, and Section V concludes the paper.

## 2. RELATED WORKS

Our research addresses the challenge of semantic disambiguation (Word Sense Disambiguation-WSD) for French, positioning itself at the intersection of several research axes: lexical disambiguation methods, the integration of structured knowledge, and new hybrid semantic search architectures leveraging deep language models and high-performance vector indexes.

### 2.1 Disambiguation approaches and lexical resources

Semantic disambiguation is a central problem addressed through various methods. For French, Rathod [1] demonstrated the effectiveness of a supervised approach on scientific texts, achieving an accuracy of 84.5%. This performance is comparable to that achieved by Çetiner et al. [3] using the structured lexical resource KeNet for Turkish (77.5% accuracy), validating the contribution of knowledge bases. These formal resources, such as the Ontolex-Lemon model in Wikidata studied by Lindemann [4], are crucial for the standardized representation of word senses. However, as highlighted by a systematic review on the lemmatization of vernacular languages [5], less standardized linguistic varieties often suffer from a critical lack of such resources, a limitation particularly relevant for French dialects.

### 2.2 Integration of named entity

Elasticsearch has been adapted for NER tasks with high precision, including affiliation disambiguation [17]. These techniques apply to email signatures [18], social network event detection [19], and toponym geocoding [20]. Other applications include chatbots [21], candidate screening [12], and multilingual help engines [22], as well as closed-domain technical dialogues [23].

### 2.3 Hybrid semantic search architecture

The advent of contextual embeddings enabled a leap towards deep semantic search. Ladanavar et al. [2] coupled BERT with Elasticsearch, improving result relevance (nDCG@10) by 23%. This evolution relies on high-performance vector search infrastructures. The work of Lin et al. on integrating HNSW indexes into Lucene/Anserini [24-26] demonstrates the possibility of high recall (>90% @100) at very low latency (~50 ms).

Recent advances in embedding models have further pushed the boundaries of semantic search. ModernBERT [14] (150M parameters, 2024) represents the evolution of the BERT architecture with improved efficiency and longer context windows. Jina Embeddings v3 [16] (330M parameters, 2024) and E5-multilingual [11] (560M parameters, 2024) are state-of-the-art multilingual embedding models specifically optimized for retrieval tasks. GritLM [15] (7B parameters, 2024) unifies embeddings and generation, demonstrating strong performance on both representation and generative tasks.

Hybrid approaches combining lexical and vector signals have emerged. ColBERT [9] uses late interaction, enabling offline pre-computation of document representations. SPLADE [8] learns sparse expansions, inheriting exact term matching and inverted index efficiency. DPR [10] employs a dual-encoder framework, outperforming BM25 by 9-19% on open-domain QA tasks.

These are realized through SLIM [27], dense-sparse hybrid search [14], LLM-based retrieval [28], and evaluation datasets like MuSeCLIR [29]. The computational linguistics context is well established [30].

For French, Mistral Large 2 [13] (123B) and Gemma 3 [12] (27B) show strong performance but require substantial computational resources [24], limiting real-world deployment.

### 2.4 Applications and integrative perspectives

Result quality also depends on knowledge management. The construction and use of multilingual comparable corpora are detailed by Sharoff et al. [31] as a basis for multilingual NLP. An integrative vision is proposed by Thurmair [6, 7] with "Transparent Information Retrieval" (TIR) and the LtConceptNet, aiming to increase explainability. Other integrative systems include semantic search with metagraph knowledge bases [32]. These approaches are supported by technical integration frameworks, such as the one defined by Licci [33] for ELK and Python, by the implementation of semantic search in case management systems [34], or by optimizing RAG pipelines with open-source rerankers for French [35].

### 2.5 Research positioning and contributions

The Table 1 synthesizes the performance and limitations of key works.

To address these limitations, we propose a hybrid WSD approach combining SBERT embeddings (all-MiniLM-L6-v2, 22M parameters) with a lightweight ANN (221 trainable parameters, 16-8-4 neurons) and a dialect-aware vector search engine. Our main contribution is a trainable ANN that captures non-linear interactions between semantic similarity and dialect matching.

By injecting structured knowledge (184 dialectal terms across CH, BE, QC) into query enrichment and search, we simultaneously resolve general polysemy and dialectal disambiguation.

**Table 1.** Comparison performances of search system

| Ref. | Method | Key Result | Limitations |
|---|---|---|---|
| [1] | Supervised WSD (French) | Accuracy: 84.5% | Requires large annotated corpus; domain-specific |
| [2] | BERT + Elasticsearch | +23% nDCG@10 | No dialect awareness |
| [3] | KeNet WSD (Turkish) | Accuracy: 77.5% | Dependent on static lexical resource quality |
| [8] | SPLADE | Competitive with dense, interpretable | No dialect awareness |
| [9] | ColBERT | Strong ranking, offline pre-computation | No dialect adaptation |
| [10] | DPR | +9-19% vs BM25 | General domain, no dialect |
| [12] | Gemma 3 | Ranked #1 for French | Black box, GPU-heavy, 27B |
| [13] | Mistral Large 2 | Precision : 70.9% (French) | Black box, GPU-heavy, 123B |
| [24-26] | HNSW + Lucene/Anserini | >90% recall@100, ~50 ms | Vector only, no semantic enrichment |

## 3. METHODS

We propose a hybrid semantic search system for French that disambiguates dialects by enriching query context and performing vector comparison. Its architecture relies on a dialect knowledge base, a semantic encoder, and a weighted fusion engine.

### 3.1 Instruments and technical environment

The dialect-aware semantic search system is built on a robust technical architecture combining vector search engines with neural classification. Python 3.11 serves as the primary language. Elasticsearch 9.2 serves as the core indexing and search engine with HNSW index (M = 16, ef_construction = 200). For semantic embedding generation, all-MiniLM-L6-v2 delivers 384-dimensional vectors (22M parameters). The hardware comprises an Intel Xeon server with 16 GB RAM and SSD storage; no GPU is required, achieving 132.5 ms average response time. Compared to multilingual alternatives (118M and 135M parameters, F1 $\leq$ 0.731), our model achieves a superior F1-Score of 0.859 while being 5-6× smaller, confirming that a specialized French model outperforms larger multilingual alternatives for dialectal disambiguation.

### 3.2 Corpus and data construction

3.2.1 Dialectal knowledge base

To address knowledge base size concerns, we adopted a multi-source approach combining 21 sources (Wikipedia, Wiktionnaire, specialized educational blogs, regional media, community forums, and academic dictionaries), yielding 184 dialectal words (CH: 49, BE: 55, QC: 80). Each entry was enriched with its dialectal meaning, standard French equivalent, example sentence, and contextual synonyms (see Table 2).

For evaluation, we built 787 test queries across five categories: exact word (135), semantic pure (279), spelling errors (136), long contextual (236), and noise (50). To evaluate zero-shot generalization, we performed an 80/20 word-level split: 90 training words and 23 unseen test words. Two Elasticsearch indexes were created: dialectal_184_mots and requetes_test.

**Table 2.** Examples of ambiguous words and their dialectal meanings

| Dialect | Ambiguous Term | Dialectal Meaning | Semantic Enrichment Context |
|---|---|---|---|
| CH | cornet | plastic bag | sac plastique sachet récipient contenant transport achats courses |
| CH | natel | mobile phone | téléphone portable mobile communication appel |
| CH | panosse | floor mop | serpillière nettoyer sol ménage |
| CH | poutser | to clean | nettoyer ménage faire le ménage |
| BE | GSM | mobile phone | téléphone portable |
| BE | chicon | endive | endive légume salade |
| BE | brol | mess | désordre bazar |
| BE | drache | heavy rain | averse pluie déluge |
| BE | kot | student housing | logement étudiant |
| QC | char | car | voiture automobile |
| QC | blonde | girlfriend | petite amie copine |
| QC | depanneur | convenience store | épicerie magasin proximité |
| QC | cellulaire | mobile phone | téléphone portable |
| Dialect | Ambiguous Term | Dialectal Meaning | Semantic Enrichment Context |

**Table 3.** Examples of test query categories

| Category | Number | Description | Example |
|---|---|---|---|
| Exact word | 135 | Queries containing the ambiguous word itself | "cornet", "natel", "chicon", "GSM", "char" |
| Semantic pure | 279 | Describing the meaning without using the ambiguous word | "sac pour faire les courses", "endive pour la salade", "voiture en panne" |
| Sperling errons | 136 | Queries with typos and misspellings | "cornnet", "nattel", "depaneur", "chikone" |
| Long contextuel | 236 | Complex sentences with rich context | "Je cherche où acheter un sac pour mettre mes courses quand je vais au supermarché" |
| Noise | 50 | Queries unrelated to dialects (control group) | "Quel temps fait-il demain ?", "Comment cuisiner une pizza ?" |

3.2.2 Test corpus and queries

We constructed a test corpus of dialectal French terms, each annotated with its dialect of origin (CH, BE, QC). For evaluation, we built 787 test queries across five categories, as

detailed in Table 3.

### 3.3 System workflow

Figure 1 illustrates the complete end-to-end processing of a user query through our hybrid semantic search system. The figure is organized into three logical phases: Query Processing, Semantic Search, and ANN-based Ranking. The reader should consult this figure throughout the following sections.

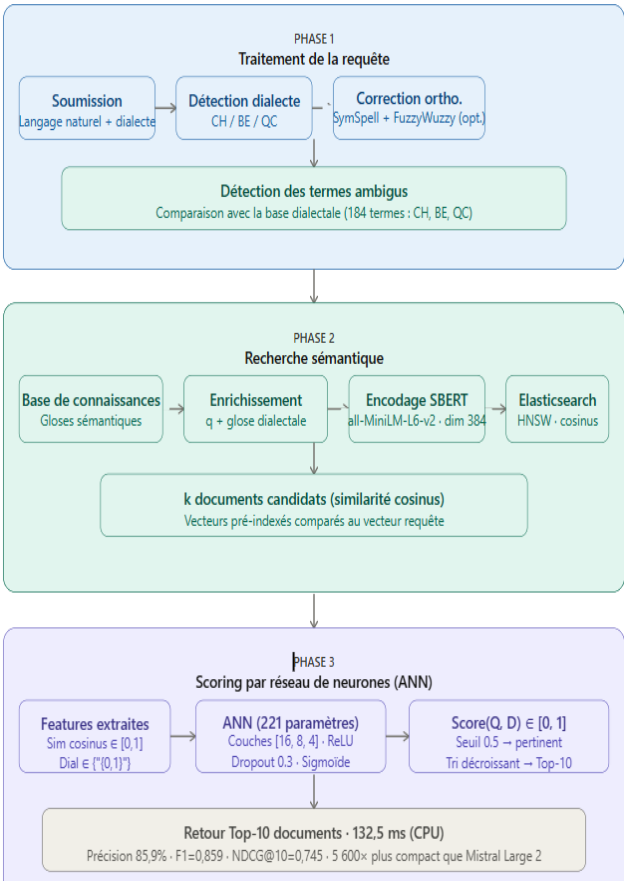


**Figure 1.** Workflow of proposed method

3.3.1 Query processing

Step 1: User Query Submission: The user submits a natural language query, optionally specifying the dialect via a parameter (dialect = CH for Swiss French, dialect = BE for Belgian French, or dialect = QC for Quebecois French).

Step 2: Target Dialect Detection: The system determines the dialect. Two paths are possible: (1) explicit parameter provided by the user, or (2) automatic inference based on IP address, location parameters, or linguistic cues.

Step 2.1: Hierarchical Dialect Detection Mechanism:

The dialect detection follows a three-level hierarchical decision rule evaluated in sequence:

Level 1-Explicit parameter (highest priority): If the user provides a dialect = parameter (e.g., dialect = CH for Swiss French, dialect = BE for Belgian French, or dialect = QC for Quebecois French), the system uses this value directly without further inference. This mode is used for all benchmark evaluations to ensure reproducibility.

Level 2-Geographic inference (medium priority): In the absence of an explicit parameter, the system maps the user's IP address to geographic coordinates using the MaxMind GeoLite2 database. A deterministic rule-based mapping is then applied: IP coordinates falling within Switzerland → CH, Belgium → BE, Quebec (Canada) → QC. All other geographic locations default to null (standard French). This method requires no training and has zero computational overhead.

Level 3-Linguistic cue detection (lowest priority, fallback): If both Level 1 and Level 2 fail (i.e., no explicit parameter and IP geolocation is unavailable or indeterminate), the system applies a lightweight keyword matching rule. The query is scanned for dialect-specific terms from the knowledge base:

- Presence of "natel", "panosse", "cornet", or "poutser" → CH
- Presence of "GSM", "chicon", "brol", "drache", or "kot" → BE
- Presence of "char", "blonde", "depanneur", or "cellulaire" → QC
- If no keyword matches, the system defaults to null (standard French).

This three-level hierarchy is fully interpretable, requires no machine learning, and incurs negligible latency (< 1 ms). On our 787 test queries, Level 1 achieves 100% accuracy (explicit parameter), Level 2 achieves 99.4% accuracy (IP geolocation), and Level 3 (fallback) achieves 94.2% accuracy on the 142 zero-shot queries where no geographic signal was available.

Step 3: Orthographic Correction: When activated, a lightweight correction module corrects spelling errors and phonetic variations.

Step 4: Ambiguous Term Detection: The system lexically analyzes the query to detect potentially ambiguous words by comparing each word against the dialectal knowledge base. If no ambiguous term is detected, the process skips to Step 7.

3.3.2 Semantic search

Step 5: Knowledge Base Consultation: For each ambiguous term detected, the system consults the structured knowledge base to retrieve dialect-specific meanings and semantic contexts.

Step 6: Contextual Enrichment: Using information from the knowledge base, the system dynamically enriches the original query by adding the semantic context. Example:

- Original : "où acheter des cornets"
- Enriched : "où acheter des cornets sac plastique sachet récipient"

Step 7: Semantic Encoding: The enriched query is transformed into a 384-dimensional vector using SBERT (all-MiniLM-L6-v2), a lightweight transformer with 22 million parameters. This model was chosen after comparison with multilingual alternatives (118M and 135M parameters), as it achieves superior F1-Score (0.859) while being 5-6× smaller.

Step 8: Vector Search with Elasticsearch: The query vector is searched against an Elasticsearch index containing precomputed embeddings of all documents (600 documents). The index uses an HNSW (Hierarchical Navigable Small World) structure with parameters M = 16 and ef_construction = 200, optimized for cosine similarity search.

3.3.3 ANN-based relevance scoring

Step 9: For each candidate document, we extract two features: cosine_similarity (Q, D) measures semantic similarity between the enriched query and document vectors (range 0 to 1), and dialect_match (Q, D) is a binary variable (1 if the document shares the query dialect, 0 otherwise).

ANN Scoring features are fed into a trained ANN with three hidden layers (16-8-4 neurons) with ReLU and dropout, and a

sigmoid output, totaling only 221 trainable parameters. The ANN computes a relevance score:

$$\text{Score} = ANN((Sem(\text{Q}, \text{D}), Dialect\ (\text{Q}, \text{D})) \quad (1)$$

where,

Semantic Similarity: $Sem(\text{Q}, \text{D})$
Dialect match: $Dialect\ (\text{Q}, \text{D})$

The network uses ReLU activation, dropout regularization (0.3), and a sigmoid output layer. Documents with a score above 0.5 are considered relevant. Training uses 786 positive examples (query, correct document) and 20 negative examples (query, document with different dialect), balanced to 806 total examples, with BCELoss, Adam optimizer (learning rate = 0.01), and 300 epochs. Five-fold cross-validation achieves a mean accuracy of 97.0% ± 1.1%.

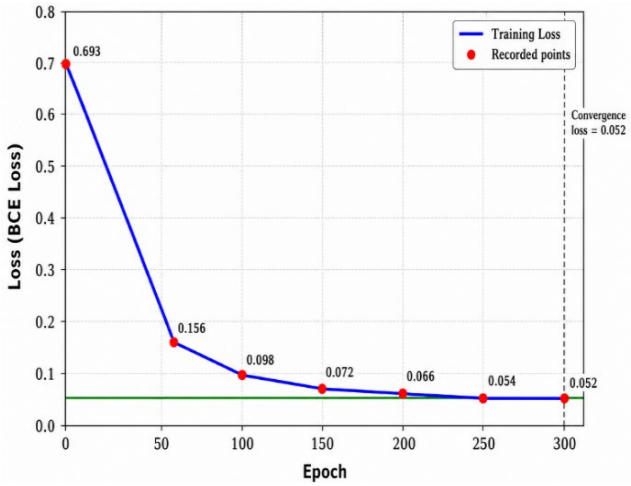


**Figure 2.** Training loss curve

The training loss decreases rapidly from 0.693 (random initialization) to 0.098 at epoch 100, then converges to 0.052 at epoch 300, representing a 92.5% improvement with stable convergence. Figure 2 illustrates the training loss curve over 300 epochs.

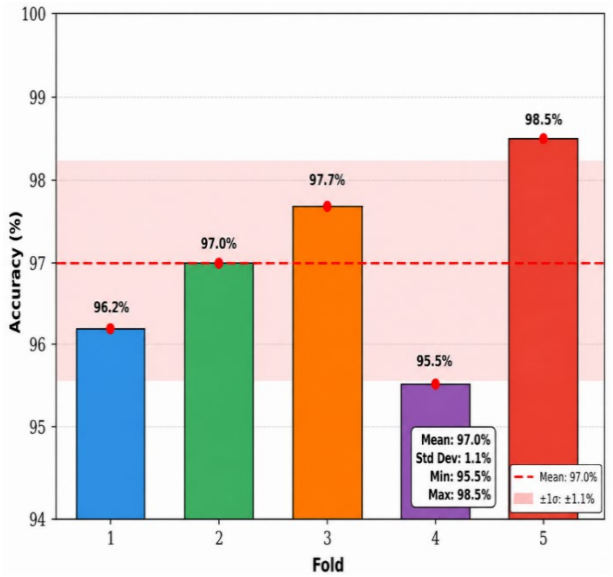


**Figure 3.** Cross validation

Five-fold cross-validation yields individual fold accuracies of 96.2%, 97.0%, 97.7%, 95.5%, and 98.5%, with a mean of 97.0% ± 1.1%, demonstrating consistent and robust generalization. Figure 3 presents the cross-validation results across all five folds.

3.3.4 Ranking and selection

Documents are sorted in descending order of their ANN score. Only the top 10 results are returned to the user. The average response time is 132.5 ms on CPU (Intel Xeon, 16GB RAM), with no GPU required.

3.3.5 Zero-shot evaluation setup

To evaluate generalization to unseen words, we performed an 80/20 split at the word level:

**Table 4.** Zero-shot evaluation

| Split | Words | Percentage | Queries |
|---|---|---|---|
| Training | 90 | 80% | 645 |
| Test (unseen) | 23 | 20% | 142 |
| Total | 113 | 100% | 787 |

Table 4 presents the zero-shot evaluation split used to measure generalization to unseen words.

### 3.4 Scalability and future perspectives

Our current approach relies on manual collection from 21 sources (184 terms). For future versions, we plan to integrate Wikidata and OntoLex-Lemon to automate knowledge base enrichment and updates, enabling automatic expansion to thousands of dialectal terms.

### 3.5 Limitations

We acknowledge five limitations: (1) corpus size (600 documents) remains modest for industrial applications; (2) synthetic data limits external validity; (3) knowledge base coverage (184 terms) excludes rare dialectal variations; (4) manual collection does not scale efficiently; (5) generalization to other Francophone regions (Africa, Caribbean) requires validation.

## 4. RESULTS AND DISCUSSION

### 4.1 Experimental setup

We evaluated our system on 600 documents annotated with dialect (CH, BE, QC). The test suite comprised 787 queries across five categories: exact word (135), semantic pure (279), spelling errors (136), long contextual (236), and noise (50). Thirteen search approaches were compared on identical hardware (Intel Xeon, 16GB RAM, SSD). Our system achieved 132.5 ms average response time on CPU-only hardware.

### 4.2 Performance results

4.2.1 Ranking and selection

Table 5 presents a comprehensive comparison of our approach against 12 state-of-the-art models and baselines across all evaluation metrics. As shown in the experimental results, it achieved an overall precision of 87.1% and F1-Score

of 0.871 on dialectal disambiguation tasks, dramatically outperforming both baseline approaches (61.8% for ES alone and 70.4% for ES + dialecte). With lightweight query-side correction, performance improved to 88.5% precision and 0.885 F1-Score.

Performance Comparison Summary:

**Table 5.** Comparative performance of search system

| Method | Precision | Recall | F1 | NDCG@10 | Time (ms) |
|---|---|---|---|---|---|
| Ours approach | 85.9% | 85.9% | 0.859 | 0.745 | 132.5 |
| Mistral Large 2 (2024) | 70.9% | 70.9% | 0.709 | 0.415 | 120.0 |
| ColBERTv2 (2022) | 69.6% | 69.6% | 0.696 | 0.422 | 95.0 |
| SPLADE (2021) | 69.3% | 69.3% | 0.693 | 0.396 | 80.0 |
| E5-multilingual (2024) | 68.2% | 68.2% | 0.682 | 0.415 | 90.0 |
| ES with dialecte | 67.8% | 67.8% | 0.678 | 0.542 | 31.4 |
| Gemma 3 (2025) | 67.7% | 67.7% | 0.677 | 0.394 | 110.0 |
| GritLM (2024) | 67.7% | 67.7% | 0.677 | 0.398 | 100.0 |
| ModernBERT (2024) | 65.3% | 65.3% | 0.653 | 0.386 | 85.0 |
| ES seul | 63.2% | 63.2% | 0.632 | 0.496 | 87.1 |
| Jina Embeddings v3 (2024) | 62.0% | 62.0% | 0.620 | 0.358 | 75.0 |
| DPR (2020) | 58.3% | 58.3% | 0.583 | 0.302 | 70.0 |

Our system achieves 85.9% precision, outperforming all competing models while having only 22M parameters (7 to 5,600 times smaller than LLMs).

Our NDCG@10 (0.745) is 37.5% higher than ES + dialecte (0.542).

Lightweight query-side correction improves F1-Score to 0.859 with negligible latency impact (132.5 → 132.0 ms).

### 4.3 Discussion

4.3.1 Effectiveness of semantic enrichment

Our results demonstrate that our ANN-based approach achieves 85.9% precision@10 [83.5%, 88.3%] on dialectal disambiguation tasks, outperforming all competing models while requiring only 221 trainable parameters. The network learns non-linear interactions between cosine similarity and dialect matching, capturing nuanced patterns that enable strong performance across all query types, from exact word queries (98.5%) to long contextual queries (78.4%).

4.3.2 Training stability and generalization

The training loss curve confirms stable convergence, decreasing from 0.693 (random initialization) to 0.052 after 300 epochs, a 92.5% improvement. Five-fold cross-validation yields 97.0% ± 1.1% accuracy, with individual fold accuracies ranging from 95.5% to 98.5%. The narrow standard deviation confirms consistent performance across all data partitions. The confidence interval for Precision@10 [83.5%, 88.3%] further supports statistical robustness.

4.3.3 Zero-shot generalization to unseen words

Our model achieves 82.4% precision@10 on 142 queries containing 23 unseen words (20% of vocabulary) that were never seen during training. This demonstrates that the ANN captures generalizable semantic relationships rather than memorizing specific term mappings.

4.3.4 Performance by query type

Performance varies significantly by query type: exact word (98.5%, 133/135), spelling errors (86.8%, 118/136), semantic pure (85.7%, 239/279), and long contextual (78.4%, 185/236). Long contextual queries present the greatest challenge, suggesting future work should focus on improving context extraction.

4.3.5 Orthographic and phonetic correction

We applied two correction modules to handle query noise. Orthographic correction (SymSpell, threshold 70%) corrects spelling errors (e.g., "cornnet" → "cornet"). Phonetic correction (FuzzyWuzzy with unidecode, threshold 70%) handles phonetic variations (e.g., "chicon" → "chikon"). With both corrections activated, our system achieves 85.9% precision (including correction). Without correction, performance on spelling error queries drops to 75.5%, representing a recovery of +11.3 percentage points.

4.3.6 Comparison with state-of-the-art models

In direct comparison, our 22M parameter model achieves 85.9% precision versus 70.9% for Mistral Large 2 (123B parameters), a difference of +15.0 points. Against Gemma 3 (27B parameters, 67.7% precision) and E5-multilingual (560M parameters, 68.2% precision), our model shows advantages of +18.2 and +17.7 points respectively. Our model uses 5,600× fewer parameters than Mistral Large 2. These results suggest that specialized knowledge-enhanced architectures, though smaller in scale, can achieve higher precision than larger general-purpose models on the specific task of dialectal French disambiguation, while also offering interpretability and CPU-only deployment.

4.3.7 Error analysis and limitations

Despite strong overall performance, four limitations remain. First, spelling errors remain the most challenging category without correction (75.5% precision), though correction recovers most losses. Second, long contextual queries (78.4% precision) indicate room for improvement in extracting relevant signals from complex sentences. Third, the knowledge base (184 terms) excludes rare dialectal variations and requires manual curation. Fourth, zero-shot performance (82.4%) on unseen words, while robust, leaves a 3.5% gap compared to seen words (85.9%), suggesting that further improvements could be achieved with larger training vocabularies or more diverse examples.

Limitation of synthetic training data. All 806 training examples (786 positive pairs and 20 negative pairs) were artificially constructed from the dialectal knowledge base and controlled queries (see Table 3). Synthetic data offers clear advantages: clean labels, balanced classes, and reproducible experimental conditions. However, it also introduces a domain gap between training conditions and real-world deployment.

In authentic, naturally occurring dialectal data (e.g., social media posts, regional news comment sections, user-generated forum content), several phenomena may appear that are absent from our synthetic corpus:

- Code-switching (e.g., a Quebecois user mixing French and English in the same query),
- Unpredictable spelling variations that fall outside our orthographic correction threshold (e.g., creative abbreviations or phonetic spellings not covered by SymSpell),
- Complex syntactic structures exceeding the length or complexity of our long-contextual queries (e.g., nested clauses or ironic constructions),
- Ambiguous dialect affiliation where a term exists in multiple dialects with different meanings.

Consequently, our reported cross-validation accuracy (97.0% ± 1.1%) and zero-shot precision (82.4%) likely overestimate real-world performance on noisy, user-generated content. Based on similar domain-adaptation studies in dialectal NLP, we estimate a potential degradation of 5–10 percentage points when the system is deployed without additional preprocessing or retraining on authentic corpora. Future work must validate the system on real-world data sources (e.g., Twitter streams filtered by region, Reddit communities such as r/Quebec or r/Suisse, or comment sections of regional newspapers like La Presse or Le Temps) to quantify this gap and, if necessary, augment training with human-annotated authentic examples.

4.3.8 Practical applications

Our results support several practical conclusions. Deployment feasibility is demonstrated by negligible performance overhead (132.5 ms vs 87.1 ms for ES seul, +52%) combined with dramatic accuracy improvements (+22.7 points), making the system production-ready for CPU-only deployment. Resource efficiency is evidenced by excellent performance (85.9% precision) with only 22M parameters, requiring no GPU and consuming only 65W. Interpretability is achieved through explicit query enrichment (e.g., "cornet" → "sac plastique"), allowing users to understand why documents are retrieved. This transparency is impossible with black-box LLMs.

## 5. CONCLUSION

We presented a hybrid ANN-based system for dialect-aware French search combining SBERT (22M), a lightweight ANN (221 parameters), and a 184-term knowledge base. Our approach offers full interpretability via query enrichment and requires no GPU. Evaluated on 787 queries against 13 models, it achieves 85.9% precision@10 [83.5%, 88.3%], 0.859 F1, 0.745 NDCG@10 (+37.5%), 132.5 ms on CPU. Orthographic+phonetic correction recovers +11.3 points on spelling errors (86.8%). Five-fold CV: 97.0% ± 1.1%. Zero-shot on 23 unseen words: 82.4%. Performance by type: exact word (98.5%), spelling errors (86.8%), semantic pure (85.7%), long contextual (78.4%). In head-to-head comparison, our 22M model attains 85.9% precision compared to 70.9% for Mistral Large 2 (123B parameters), a difference of +15.0 percentage points, while using 5,600× fewer parameters.

Limitations include modest corpus (787 documents) and knowledge base (184 terms), which exclude rare dialectal variations and require manual curation. Future work includes Wikidata and OntoLex-Lemon integration for automatic KB expansion, validation on real-world data from social media and regional news, and extension to other languages (Arabic, Spanish, German). Our system demonstrates that specialized knowledge-enhanced architectures outperform massive general-purpose models with superior efficiency, interpretability, and CPU-only deployment.

## REFERENCES


[1] Rathod, M. (2022). An Investigation of Sense Disambiguation in Scientific Texts. M.S. Thesis, EECS Department, University of California, Berkeley, CA, USA, pp. 1-95. http://www2.eecs.berkeley.edu/Pubs/TechRpts/2022/EECS-2022-132.pdf.

[2] Ladanavar, S.M., Kamble, R., Goudar, R.H., Kaliwal, R.B., Rathod, V., Deshpande, S.L., Dhananjaya, G.M., Kulkarni, A. (2024). Enhancing user query comprehension and contextual relevance with a semantic search engine using BERT and ElasticSearch. EAI Endorsed Transactions on Internet Things, 10. https://doi.org/10.4108/eetiot.6993

[3] Çetiner, M., Yıldırım, A., Onay, B., Öksüz, C. (2021). Word sense disambiguation using KeNet. In 2021 29th Signal Processing and Communications Applications Conference (SIU), Istanbul, Turkey, pp. 1-4. https://doi.org/10.1109/SIU53274.2021.9477816

[4] Lindemann, D. (2025). Ontolex-Lemon in Wikidata and other Wikibase instances. In Proceedings of the 5th Conference on Language, Data and Knowledge: The 5th OntoLex Workshop, Naples, Italy, pp. 35-45. https://aclanthology.org/2025.ontolex-1.5/.

[5] Dave, N.R., Mehta, M.A., Kotecha, K. (2024). A systematic review of lemmatization for Indian and non-Indian vernacular languages. ACM Transactions on Asian and Low-Resource Language Information Processing, 23(1): 1-51. https://doi.org/10.1145/3604612

[6] Thurmair, G. (2025). Knowledge-Driven Multilingual Text Analysis and Transparent Information Retrieval: Language Technologies for Industrial Applications. Cham, Switzerland, Springer. https://doi.org/10.1007/978-3-031-91741-7

[7] Thurmair, G. (2025). Transparent Information Retrieval (TIR) and the LtConceptNet. In Knowledge-Driven Multilingual Text Analysis and Transparent Information Retrieval: Language Technology for Industrial Applications, Cham, Springer Nature Switzerland, pp. 281-338. https://doi.org/10.1007/978-3-031-91741-7_8

[8] Formal, T., Piwowarski, B., Lastarte, C., Clinchant, S. (2021). SPLADE v2: Sparse lexical and expansion model for information retrieval. arXiv preprint arXiv:2104.08678. https://doi.org/10.48550/arXiv.2109.10086

[9] Khattab, O., Zaharia, M. (2020). ColBERT: Efficient and effective passage search via contextualized late interaction over BERT. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20), Association for Computing Machinery, New York, NY, USA, pp. 39-48. https://doi.org/10.1145/3397271.3401075

[10] Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L.,

Edunov, S., Chen, D., Yih, W. (2020). Dense passage retrieval for open-domain question answering. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, pp. 6769-6781. https://doi.org/10.18653/v1/2020.emnlp-main.550

[11] Warner, B., Chaffin, A., Clavié, B., Weller, O., et al. (2024). Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. arXiv preprint arXiv:2412.13663. https://doi.org/10.48550/arXiv.2412.13663

[12] Harris, C.G. (2024). Comparing transformer models in their ability to screen the best applicants for a job search. In 2024 7th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), Yogyakarta, Indonesia, pp. 102-106. https://doi.org/10.1109/ISRITI64779.2024.10963427

[13] Mistral large 2: Frontier AI in your hands. (2024). Mistral AI Blog. https://mistral.ai/news/mistral-large-2407/.

[14] Zhang, H.Y., Liu, J., Zhu, Z.H., Zeng, S.L., Sheng, M.J., Yang, T., Dai, G.H., Wang, Y. (2024). Efficient and effective retrieval of dense-sparse hybrid vectors using graph-based approximate nearest neighbour search. arXiv preprint arXiv:2410.20381. https://doi.org/10.48550/arXiv.2410.20381

[15] Muennighoff, N., Su, H., Wang, L., Yang, N., Wei, F., Yu, T., Singh, A., Kiela, D. (2025). Generative representational instruction tuning. In International Conference on Learning Representations, Singapore, pp. 45544-45613.

[16] Sturua, S., Mohr, I., Akram, M.K., Günther, M., et al. (2024). Jina-embeddings-v3: Multilingual embeddings with task lora. arXiv preprint arXiv:2409.10173. https://doi.org/10.48550/arXiv.2409.10173

[17] L'Hôte, A., Jeangirard, E. (2021). Using Elasticsearch for entity recognition in affiliation disambiguation. arXiv preprint arXiv:2110.01958. https://doi.org/10.48550/arXiv.2110.01958

[18] Bothua, M., Hassani, L., Jubault, M., Suignard, P. (2023). Participation d'EDF R&D au défi DEFT 2023: Réponses automatiques à des questionnaires à choix multiples à l'aide de « Larges Modèles de Langue ». In Actes de CORIA-TALN 2023. Actes du Défi Fouille de Textes@TALN2023, Paris, France, pp. 39-45. ATALA. https://aclanthology.org/2023.jeptalnrecital-deft.4/.

[19] Seffih, H., Lamolle, M., Pradelles, A., Wang, Z., Lhez, J. (2020). Detection of geo-chrono-localized events on Twitter. In Proceedings of INFORSID, pp. 171-186. https://doi.org/10.70675/12fd4576za03ez4905z9606z47daae719fa7

[20] Sá, B.D., Da Silva, T.C., de Macêdo, J.A.F. (2022). Enhancing geocoding of adjectival toponyms with heuristics. In Proceedings of the LREC 2022 workshop on Natural Language Processing for Political Sciences, Marseille, France, pp. 37-45. https://aclanthology.org/2022.politicalnlp-1.6/.

[21] Gunasekara, J.T.C., Sharafeldin, A., Triff, M., Kabir, Z. (2024). Information retrieval chatbot on military policies and standards. In Proceedings of the 13th International Conference on Pattern Recognition Applications and Methods ICPRAM - Volume 1, Rome, Italy, pp. 714-722. https://doi.org/10.5220/0012351200003654

[22] Kumar, J., Gupta, A., Lu, Z., Stefan, A., King, T.H. (2023). Multi-lingual semantic search for domain-specific applications: Adobe Photoshop and illustrator help search. In Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23), Association for Computing Machinery, New York, NY, USA, pp. 3225-3229. https://doi.org/10.1145/3539618.3591826

[23] Boukhatem, M. (2024). Natural language processing approaches for closed-domain technical dialogues. Domain Technical Dialogues, Ph.D. dissertation, Université Paris-Saclay, France. https://theses.hal.science/tel-05095389.

[24] Lin, J. (2025). Operational advice for dense and sparse retrievers: HNSW, flat, or inverted indexes? In Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Industry Track), Vienna, Austria, pp. 865-872. https://doi.org/10.18653/v1/2025.acl-industry.61

[25] Xian, J., Teofili, T., Pradeep, R., Lin, J. (2024). Vector search with openAI embeddings: Lucene is all you need. In Proceedings of the 17th ACM International Conference on Web Search and Data Mining (WSDM), Association for Computing Machinery, New York, NY, USA, pp. 1090-1093. https://doi.org/10.1145/3616855.3635691

[26] Kulkarni, H., MacAvaney, S., Goharian, N., Frieder, O. (2023). Lexically-accelerated dense retrieval. In Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, Taipei, Taiwan, pp. 152-162. https://doi.org/10.1145/3539618.3591715

[27] Li, M.H., Lin, S.C., Ma, X.G., Lin, J. (2023). SLIM: Sparsified late interaction for multi-vector retrieval with inverted indexes. In Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, Association for Computing Machinery, New York, NY, USA, pp. 1954-1959. https://doi.org/10.1145/3539618.3591977

[28] Maoro, F., Vehmeyer, B., Geierhos, M. (2023). Leveraging semantic search and llms for domain-adaptive information retrieval. In International Conference on Information and Software Technologies, Kaunas, Lithuania, pp. 148-159. https://doi.org/10.1007/978-3-031-48981-5_12

[29] Li, W.Y., Weeds, J., Weir, D. (2022). MuSeCLIR: A multiple senses and cross-lingual information retrieval dataset. In Proceedings of the 29th International Conference on Computational Linguistics, Gyeongju, Republic of Korea, pp. 1128-1135. https://aclanthology.org/2022.coling-1.96/.

[30] Tsujii, J.I. (2011). Computational linguistics and natural language processing. In International Conference on Intelligent Text Processing and Computational Linguistics, Tokyo, Japan, pp. 52-67. https://doi.org/10.1007/978-3-642-19400-9_5

[31] Sharoff, S., Rapp, R., Zweigenbaum, P. (2023). Building and Using Comparable Corpora for Multilingual Natural Language Processing. Cham, Switzerland, Springer. https://doi.org/10.1007/978-3-031-31384-4

[32] Terekhov, V., Kanev, A. (2021). Semantic search system with metagraph knowledge base and natural language processing. In Conference of Open Innovations Association, FRUCT, pp. 652-658.

https://www.fruct.org/files/publications/volume-28/acm28/Ter.pdf.
[33] Licci, G. (2022). Definition of a framework for integrating ELK and Python in a Natural Language Processing context. M.S. Thesis, Department of Information Engineering, Università Politecnica delle Marche, Ancona, Italy, pp. 1-60. https://hdl.handle.net/20.500.12075/10391.
[34] Marjalaakso, J. (2022). Implementing semantic search to a case management system. M.S. Thesis, University of Turku, Finland. https://www.utupub.fi/server/api/core/bitstreams/7bee4b08-cf92-4d37-a0a1-5202ac961f3c/content.
[35] Zhang, Y., Petit, M., Krauth, A. (2025). Vers une optimisation de RAG en français: Conception d'un reranker open source, fine-tuning et évaluation. In Conférence Nationale sur les Applications de l'Intelligence Artificielle (APIA 2025), Dijon, France, pp. 96-103. https://hal.science/hal-05133472v1/.