Text2AQL: From Clinical Questions to Executable openEHR Queries

Marko Zeman
Better Ltd
marko.zeman@better.care

Matic Bernik
Better Ltd
matic.bernik@better.care

Robert Tovornik
Better Ltd
robert.tovornik@better.care

Borut Fabjan
Better Ltd
borut.fabjan@better.care

***Abstract*—Retrieving structured data from openEHR repositories requires proficiency in the Archetype Query Language (AQL), posing significant challenge for clinicians and healthcare professionals. To address this, we introduce Text2AQL, an AI-powered assistant that translates natural language queries into executable AQL statements. Our method combines a large language model (LLM) with Retrieval-Augmented Generation (RAG) to map clinical terms to relevant openEHR archetypes and generate syntactically and semantically accurate queries. A dataset of 1,274 AQL queries supports evaluation, ensuring that generated outputs align closely with clinical intent. Preliminary results show that Text2AQL improves the general-purpose LLMs in archetype identification, matching data element and path accuracy. By enabling clinicians to ask data questions in plain language, we can democratize access to healthcare data, enhance clinical decision-making and streamline research workflows.**

***Index Terms*—Text2AQL, Archetype Query Language (AQL), Natural Language Processing (NLP), Artificial Intelligence (AI), openEHR, Retrieval-Augmented Generation (RAG)**

I. Introduction

OpenEHR [2] has become a prominent standard for long-term healthcare data persistence and interoperability, facilitating structured clinical data management. Retrieving data from openEHR repositories requires the use of AQL - Archetype Query Language, a powerful yet complex querying language [7]. Mastery of AQL involves understanding its syntax, grammar and the underlying openEHR clinical models, which hinders adoption. Typically, AQL queries are manually crafted, a process demanding substantial time, expertise and familiarity with clinical archetypes. Although semi-automated approaches with autocomplete functionality and template-based construction exist, they only partially alleviate the complexities associated with query formulation.

A notable advantage of the openEHR standard lies in its use of formally described clinical models (archetypes and templates), which makes the structure of clinical data more navigable and predictable than in traditional SQL-based systems.

Our research addresses this challenge by developing an AI-driven Text2AQL assistant capable of translating natural language queries into executable AQL statements. Clinicians routinely seek answers to clinical questions like "*How long has the patient X been taking Levaquin?*" or "*Does patient X have an allergy to penicillin?*". By enabling users to effortlessly input such questions in everyday language, we aim to significantly accelerate clinical data retrieval, democratizing data access for clinicians, healthcare professionals and even patients.

II. Related work

Our work is closely related to the field of Text2SQL translation, which focuses on converting natural language questions into structured database queries. Pioneering efforts such as Seq2SQL [14] and SQLNet [12] introduced neural architectures capable of learning to generate SQL queries from annotated question-query pairs. These approaches were further advanced by models like RAT-SQL [11], which incorporate schema representations and attention mechanisms to improve generalization across different databases.

Recent advances leverage large language models (LLMs), such as GPT-3 [3], GPT-4 [1] and GPT-5 [6], which demonstrate strong capabilities in generating SQL queries with little to no fine-tuning. Despite strong performance, these models often struggle with domain-specific constraints, hallucinate elements and lack grounding in real-world data models, which are particularly critical challenges in the clinical domain.

In the healthcare context, several prior works have explored LLMs for clinical data [9], [10], [13], [15], yet we are unaware of approaches that directly address the complexities of AQL and openEHR.

III. Methods

To enable the translation of natural language into executable AQL queries, our approach follows a multi-step process presented in Figure 1.

The steps of our generation process are the following:

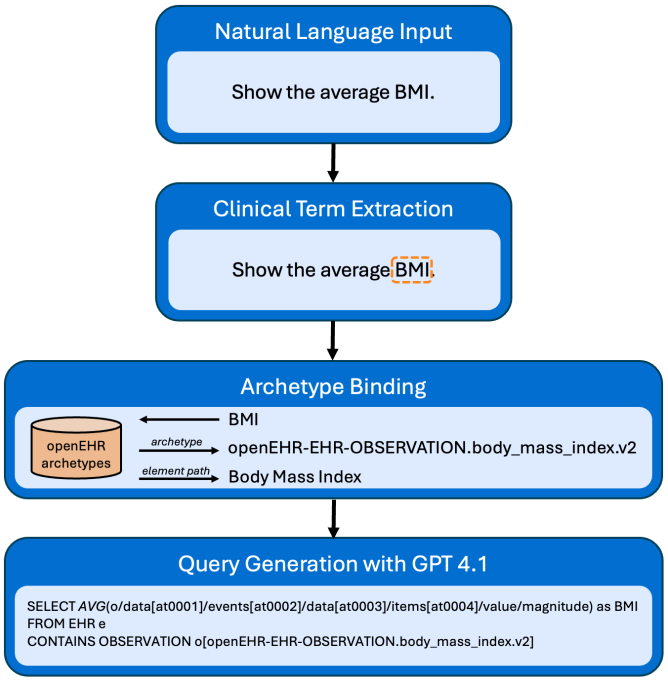


Fig. 1: Text2AQL generation process.

A. Clinical Term Extraction

Natural language queries provided by users are analyzed to extract key clinical concepts required for AQL formulation.

B. Archetype Binding

We map identified clinical terms to relevant openEHR archetype data elements using RAG – Retrieval-Augmented Generation [5], leveraging a vector database implemented in OpenSearch [8]. This improves semantic matching by retrieving the most relevant archetypes and element paths based on contextual similarity.

C. Query Generation

Exploiting structural parallels between AQL and SQL, our system utilizes the LLM, specifically GPT-4.1, to propose candidate AQL queries with accurate archetypes and their elements' paths. These candidates are then passed through a syntax validation procedure, which detects and corrects errors before producing the final AQL query.

IV. Experimental Setup

Post-generation, the AQL queries are assessed for correctness using a benchmark dataset comprising 1,274 natural language descriptions paired with validated AQL queries. To estimate the similarity between generated outputs and ground truth, we adopt the *LLM-as-a-judge* paradigm [4], employing GPT-4o to numerically evaluate semantic similarity.

The input to GPT-4o (our *LLM-as-a-judge*) is a pair consisting of the ground truth AQL and a generated AQL. The model outputs a numerical semantic similarity score (ranging from 1 – completely dissimilar to 5 – identical), along with a textual justification of the score, which we use to gain deeper understanding.

In addition, we evaluated the syntactic correctness of the generated AQLs, classifying each as either syntactically valid or invalid.

V. Results

Large language models (LLMs) such as those from the GPT family are typically trained on large corpora of general-purpose SQL queries, but have limited or no exposure to AQL queries. As a result, these models frequently conflate SQL and AQL syntax, leading to the generation of invalid AQL statements. Common errors include the incorrect use of SQL constructs like `SELECT *`, improper structuring of `CONTAINS` clauses without maintaining correct parent-child relationships and hallucination of non-existent paths within archetypes. General-purpose LLMs, when applied without domain-specific enhancements, frequently produce invalid AQL queries. Figure 2 presents two examples of such incorrect outputs.

Natural Language Input:
List unique template IDs.

General-purpose LLM (GPT-4o)

```
SELECT DISTINCT template_id
FROM EHR e
CONTAINS COMPOSITION c
```

Text2AQL

```
SELECT DISTINCT c/archetype_details/
    template_id/value AS template_id
FROM EHR e
CONTAINS COMPOSITION c
```

Natural Language Input:
Count the number of all EHR IDs.

General-purpose LLM (GPT-4o)

```
SELECT COUNT(ehr_id) AS ehr_count
FROM EHR
```

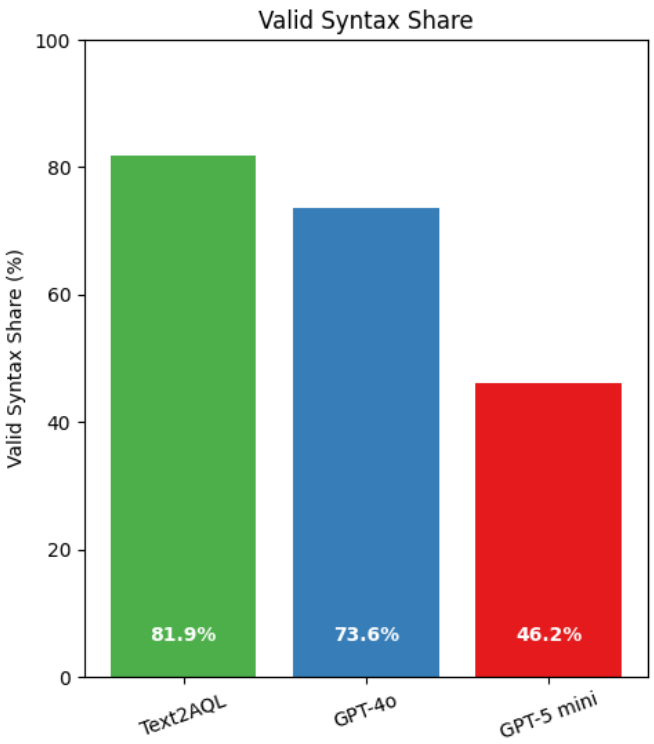
Text2AQL

```
SELECT COUNT(e/ehr_id/value) AS ehr_count
FROM EHR e
```

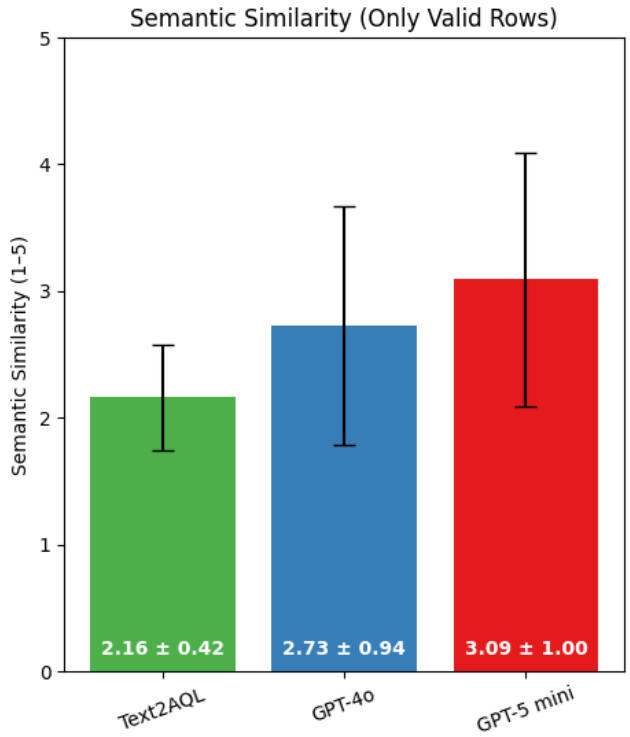
Fig. 2: Comparison of AQL queries generated by general-purpose LLM vs. our Text2AQL.

Next, we present a comparison between GPT-4o, GPT-5 mini and our Text2AQL generation from a natural language input, i.e. ground truth AQL description.

In Figure 3a, we compare the syntactic validity of generated AQLs across the three approaches. Text2AQL achieves the



(a) Syntax validity of generated AQLs.



(b) Semantic similarity to the ground truth AQL, evaluated only on valid queries using *LLM-as-a-judge* numerical evaluation.

Fig. 3: Comparison between approaches regarding (a) syntax validity and (b) semantic similarity.

highest share of syntactically valid queries (81.9%), outperforming both GPT-4o (73.6%) and GPT-5 mini (46.2%). This demonstrates that our method is considerably more reliable in producing queries that can be executed without errors.

Figure 3b shows the semantic similarity to the ground truth AQL considering only syntactically valid queries. Here, GPT-5 mini reaches the highest average similarity score (3.09 ± 1.00), followed by GPT-4o (2.73 ± 0.94), while Text2AQL achieves 2.16 ± 0.42. It is important to note, however, that both GPT-4o and GPT-5 mini produced substantially fewer valid queries overall (as shown in Fig. 3). Therefore, although the valid outputs from GPT-based methods show higher semantic alignment, Text2AQL remains the most effective approach for consistently generating a larger set of executable queries.

Each method shows distinct strengths: Text2AQL is the strongest in terms of syntactic validity, consistently generating the largest pool of executable queries, while GPT-5 mini achieves the highest semantic similarity to the ground truth AQL, but only on the smaller subset of its valid outputs.

VI. Conclusion

Our Text2AQL assistant represents an advancement in bridging the gap between complex openEHR-based clinical data structures and intuitive natural language interaction. By leveraging advanced AI and NLP methodologies, including context-aware language understanding and Retrieval-Augmented Generation, the system effectively converts free-text clinical queries into executable AQL queries. This not only streamlines clinical data retrieval but also enhances accessibility for users without specialized technical expertise.

Key strengths of the approach include automated query generation and integration with openEHR repositories while maintaining data security. Moreover, the assistant supports query refinement through explanatory feedback and troubleshooting capabilities, thereby improving both reliability and user engagement. Overall, Text2AQL offers a scalable and practical solution for facilitating efficient and user-friendly access to structured healthcare data.

Future work will focus on enhancing robustness through extensive testing on synthetic patient data and expanding curated validation datasets. Further improvements include refining clinical term extraction, increasing RAG accuracy and enforcing AQL generation in strict adherence to specification syntax rules, ultimately improving reliability of the system.

REFERENCES

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. GPT-4 Technical Report, 2023.

[2] Thomas Beale and Sam Heard. The openEHR Foundation. *Studies in Health Technology and Informatics*, 129:153–157, 2007.

[3] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[4] Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, et al. A Survey on LLM-as-a-Judge. *arXiv preprint arXiv:2411.15594*, 2024.

[5] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Igor Kulikov, Angela Fan, Vishrav Chaudhary, Ahmed El-Kishky, et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 9459–9474, 2020.

[6] OpenAI. GPT-5. https://openai.com/index/introducing-gpt-5-for-developers/, 2025. Accessed: 2025-08-25.

[7] openEHR Foundation. Archetype Query Language (AQL) Specification. https://specifications.openehr.org/releases/QUERY/latest/AQL.html, 2020. Accessed: 2025-07-30.

[8] OpenSearch Project. OpenSearch: Community-driven, open source search and analytics suite. https://opensearch.org/, 2021. Accessed: 2025-07-30.

[9] Walid Saba, Suzanne Wendelken, and James Shanahan. Question-Answering Based Summarization of Electronic Health Records using Retrieval Augmented Generation. *arXiv preprint arXiv:2401.01469*, 2024.

[10] Jerrin John Thomas, Pruthwik Mishra, Dipti Sharma, and Parameswari Krishnamurthy. LTRC-IIITH at EHRSQL 2024: Enhancing Reliability of Text-to-SQL Systems through Abstention and Confidence Thresholding. In *Proceedings of the Clinical NLP Workshop at NAACL-2024*, 2024.

[11] Bailin Wang, Richard Shin, Xiaodong Lin, Oleksandr Polozov, and Matthew Richardson. RAT-SQL: Relation-Aware Schema Encoding and Linking for Text-to-SQL Parsers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7567–7578. Association for Computational Linguistics, 2020.

[12] Xiaojun Xu, Chang Liu, and Dawn Song. SQLNet: Generating Structured Queries from Natural Language Without Reinforcement Learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 681–691. Association for Computational Linguistics, 2017.

[13] Yuanhao Yang, Selin Tarlaci, Omar Elish, Parikshit Sondhi, Nikhil Gupta, Mo Yu, and Meng Jiang. Language Models as Clinical Knowledge Workers. In *Proceedings of the Conference on Health, Inference, and Learning*, pages 279–290. PMLR, 2022.

[14] Victor Zhong, Caiming Xiong, and Richard Socher. Seq2sql: Generating structured queries from natural language using reinforcement learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 959–970. Association for Computational Linguistics, 2017.

[15] Angelo Ziletti and Leonardo D'Ambrosi. Retrieval augmented text-to-SQL generation for epidemiological question answering using electronic health records. *arXiv preprint arXiv:2403.09226*, 2024.