Adaptive Risk Scoring Queries for Longitudinal Student Mental Health Data: A Pre-Proposal

Hemit Rana[1], Sarvesh Solanke[1], Charmiben Patel[1], Chetan Thakur[1],
Jasmeen Kaur[1], Harpreet Singh[1], Anurag Sharma[1]
[1]University of Windsor, Windsor, ON, Canada
rana6c@uwindsor.ca, solanke1@uwindsor.ca, patel4mc@uwindsor.ca, thakurc@uwindsor.ca,
jasmeen1@uwindsor.ca, singh434@uwindsor.ca, sharm2u1@uwindsor.ca

***Abstract*—This pre-proposal outlines a database systems project investigating how evolving analytical semantics in risk-scoring queries can be executed efficiently over longitudinal relational and vector databases. Rather than assuming static query definitions, we explicitly model and evaluate how different execution strategies respond to frequent changes in scoring logic, revealing trade-offs in latency, storage, and update costs that are not captured in existing benchmarks.**

I. Core Idea

Universities increasingly deploy longitudinal well-being surveys to identify at-risk students and guide interventions. These analytics rely on complex risk-scoring queries that aggregate stress indicators, diagnostic scores, and temporal trends over student data. The fundamental challenge is that risk definitions are not static: thresholds, weights, and contributing factors are frequently revised as institutions update policies or gain new insights. While traditional relational databases excel at executing fixed queries, they implicitly assume stable analytical semantics and offer limited support for queries whose logic evolves. This project studies how to express and execute adaptive risk-scoring queries declaratively over a polyglot database combining structured relational data and unstructured embeddings, while systematically evaluating multiple execution strategies (baseline SQL recomputation, materialized views, incremental computation, and window-function-centric approaches) on real student mental health data to reveal non-obvious performance trade-offs.

II. Novelty Claim

Existing database research has extensively studied incremental view maintenance, temporal analytics, and materialized view selection, but these works assume that the query definition itself remains fixed and focus instead on handling data updates. Modern analytical workloads in domains such as student well-being, finance, and policy analysis challenge this assumption: the analytical definitions themselves evolve frequently as practitioners revise risk criteria or incorporate new insights. Prior work on adaptive query processing addresses runtime variability in data or resources, not semantic evolution of analytical intent. This project explicitly models evolving risk-scoring semantics as a first-class workload characteristic and experimentally demonstrates that different execution strategies exhibit dramatically different performance profiles under changing query logic. Our polyglot approach, combining relational aggregation with vector similarity search over embedded text responses, reflects modern analytical systems that span multiple data models. By conducting a systematic comparative evaluation on a real student mental health dataset, we reveal which execution strategies are most responsive to semantic changes and under what conditions each approach is optimal insights that are not captured in existing benchmarks and that can inform the design of future database optimizers and materialization strategies.

III. Related Work

The closest related work is Gupta and Mumick's foundational study on materialized view maintenance [1], which establishes core techniques for efficiently recomputing views after data changes. However, their work assumes that the query definition (the view definition) remains constant; in contrast, we study how view maintenance strategies perform when the scoring logic itself changes frequently. Nikolic et al.'s work on incremental view maintenance for analytical queries [2] advances these techniques with sophisticated delta computation strategies, but similarly assumes stable query semantics and focuses on data-driven updates rather than changes in analytical intent.

Leis et al.'s empirical study on query optimizer limitations [3] reveals that modern optimizers often fail under non-standard workloads and assumptions, motivating our focus on a workload characteristic (evolving analytical semantics) that current optimizers do not explicitly handle. Their findings suggest that existing cost models and optimization strategies are brittle under departures from static-query assumptions.

Research on temporal and window-based analytics, including foundational work by Arasu et al. on STREAM [4], addresses efficient processing of continuous queries over data streams, but assumes the query logic is fixed; the novelty in that work concerns handling streaming data arrival, not semantic evolution of the analytical queries themselves. Similarly, Babu and Widom's research on continuous queries [5] introduces adaptive processing ideas to handle runtime uncertainty, but again the query definition remains constant.

Boncz et al.'s work on MonetDB/X100 [6] demonstrates how execution-level optimizations can significantly improve performance through vectorization and pipelining. Our evaluation will use such execution models as a baseline to understand whether strategy-level differences (recomputation vs. materialization vs. incremental) dominate over execution-model differences.

Psaroudakis et al.'s systematic study of adaptive query processing [7] focuses on systems that adjust execution plans at runtime based on observed data characteristics or resource availability, an orthogonal form of adaptivity compared to changes in analytical intent. Kersten et al.'s recent comparison of compiled and vectorized query execution [8] provides a valuable execution perspective but does not consider how different execution strategies respond to semantic evolution of queries.

Our work is novel in explicitly centering on evolving risk-scoring queries as a first-class workload concern and systematically comparing execution strategies (SQL recomputation, materialized views, incremental computation, window functions) under this workload model. Unlike prior work that assumes fixed analytical semantics, we experimentally demonstrate trade-offs in latency, storage, and update costs that emerge specifically when scoring logic evolves at different frequencies. The polyglot design combining structured SQL analytics with vector similarity over embedded text reflects the hybrid nature of modern analytical systems and has not been systematically evaluated in the context of evolving scoring semantics.

References

[1] A. Gupta and I. S. Mumick, "Maintenance of materialized views: Problems, techniques, and applications," *Bulletin of the IEEE Computer Society, Technical Committee on Data Engineering*, vol. 18, no. 2, pp. 3–18, 1995.

[2] M. Nikolic, M. Olteanu, and J. Wen, "Incremental view maintenance for analytical queries," in *Proc. VLDB*, 2014, vol. 8, no. 1, pp. 25–36.

[3] V. Leis, A. Gubichev, A. Mirchev, P. Boncz, and A. Kemper, "How good are query optimizers, really?" in *Proc. VLDB*, 2015, vol. 9, no. 3, pp. 204–213.

[4] A. Arasu, S. Babu, and J. Widom, "STREAM: The Stanford data stream management system," in *Proc. SIGMOD*, 2006, pp. 665–665.

[5] S. Babu and J. Widom, "Continuous queries over data streams," *ACM SIGMOD Record*, vol. 30, no. 3, pp. 109–120, 2001.

[6] P. Boncz, M. Zukowski, and N. Nes, "MonetDB/X100: Hyper-pipelining query execution," in *Proc. CIDR*, 2005, pp. 225–237.

[7] I. Psaroudakis, T. Scheuer, and N. May, "Adaptive query processing on RAW data," in *Proc. SIGMOD*, 2016, pp. 993–1008.

[8] T. Kersten, V. Leis, T. Neumann, A. Pavlo, P. Boncz, and J. L. Kipf, "Everything you always wanted to know about compiled and vectorized queries but were afraid to ask," in *Proc. VLDB*, 2018, vol. 11, no. 13, pp. 2209–2222.