

Perancangan Arsitektur Business Intelligence Masa Depan Berbasis TurboQuant untuk Efisiensi Memori Big Data

Imam Syafi'i^{1)*} , Zainal Arif²⁾ 

¹⁾³⁾ Universitas Madura, Pamekasan, Indonesia

¹⁾ijamasyafi111@gmail.com, ²⁾arif2073@gmail.com

Abstract

Perkembangan Big Data yang semakin pesat menuntut hadirnya arsitektur Business Intelligence (BI) generasi baru yang mampu menangani volume data sangat besar dengan efisiensi memori tinggi. Sistem BI tradisional kerap menghadapi kendala serius berupa konsumsi memori berlebih serta keterbatasan kecepatan pemrosesan data berskala besar. Penelitian ini bertujuan merancang arsitektur Business Intelligence masa depan berbasis TurboQuant, yaitu kerangka arsitektural baru yang mengintegrasikan teknik kompresi data kuantitatif, pemrosesan paralel berlapis, serta manajemen memori adaptif untuk mengoptimalkan pengelolaan Big Data.

Metode penelitian mencakup perancangan arsitektur sistem TurboQuant secara konseptual, simulasi alur pemrosesan data pada berbagai skala volume Big Data, pengembangan model Data Warehouse modern berbasis skema multidimensi, serta analisis komparatif kinerja terhadap arsitektur BI konvensional. Hasil perancangan menunjukkan bahwa arsitektur TurboQuant mampu mengurangi konsumsi memori hingga 68% dibandingkan sistem BI tradisional, meningkatkan kecepatan kueri sebesar 4,3 kali lipat pada dataset bervolume tinggi, serta mendukung pemrosesan data real-time dan near real-time dengan latensi yang sangat rendah.

Arsitektur TurboQuant terbukti memberikan skalabilitas tinggi dan efisiensi penyimpanan yang signifikan, sehingga menjadikannya solusi yang relevan dan prospektif untuk kebutuhan Business Intelligence industri masa depan. Penelitian ini diharapkan dapat berkontribusi pada pengembangan sistem BI generasi berikutnya yang mampu menghadapi tantangan pengelolaan Big Data secara lebih efektif dan efisien.

Keywords: Business Intelligence, TurboQuant, Big Data, efisiensi memori, arsitektur data, Data Warehouse, pemrosesan real-time.

Article history: Received 5 April 2026, first decision 22 April 2026, accepted 22 August 2026, available online 28 October 2026

I. PENDAHULUAN

Era Big Data ditandai oleh pertumbuhan volume data yang eksponensial, di mana organisasi modern menghadapi tantangan pengelolaan data dalam skala petabyte hingga exabyte. Fenomena ini mendorong kebutuhan mendesak terhadap sistem pemrosesan data yang tidak hanya cepat, tetapi juga efisien dalam penggunaan sumber daya komputasi, khususnya memori [1], [2], [3]. Berbagai sektor industri kini sangat bergantung pada kemampuan analisis data skala besar sebagai fondasi utama dalam pengambilan keputusan strategis. Salah satu elemen penting yang perlu diperhatikan adalah data penjualan, karena informasi ini mencerminkan performa bisnis sekaligus menjadi alat ukur utama dalam menilai keuntungan perusahaan [4], [5].

Sistem Business Intelligence tradisional yang ada saat ini menghadapi keterbatasan fundamental dalam menghadapi beban kerja Big Data. Arsitektur BI konvensional dirancang pada era ketika volume data masih terbatas, sehingga mekanisme penyimpanan dan pemrosesan yang digunakan tidak dioptimalkan untuk efisiensi memori pada skala data yang sangat besar [6], [7]. Akibatnya, implementasi BI pada lingkungan Big Data sering kali membutuhkan investasi infrastruktur yang sangat besar, dengan konsumsi memori yang tidak proporsional terhadap nilai analitik yang dihasilkan. Kondisi ini mendorong kebutuhan mendesak akan arsitektur Business Intelligence baru yang dirancang dari awal dengan prinsip efisiensi memori sebagai prioritas utama [8], [9].

* Corresponding author

Kebutuhan akan arsitektur Business Intelligence masa depan yang mampu menangani tantangan Big Data mendorong lahirnya berbagai pendekatan baru dalam desain sistem analitik. Salah satu area inovasi yang paling kritis adalah optimasi penggunaan memori, di mana teknik-teknik seperti kompresi data in-memory, indexing adaptif, dan pemrosesan columnar telah terbukti mampu meningkatkan efisiensi secara signifikan. Integrasi Data Warehouse modern dengan kemampuan pemrosesan real-time dan near real-time menjadi komponen esensial dalam ekosistem BI generasi berikutnya[10], [11], [12].

Dalam konteks tersebut, penelitian ini memperkenalkan konsep TurboQuant sebagai arsitektur Business Intelligence baru yang dirancang khusus untuk mengatasi permasalahan efisiensi memori pada Big Data. TurboQuant mengadopsi pendekatan kuantisasi data yang terinspirasi dari teknik kompresi sinyal digital, dikombinasikan dengan lapisan manajemen memori adaptif dan mesin pemrosesan analitik paralel[13], [14], [15]. Arsitektur ini dirancang untuk mendukung analisis multidimensi skala besar, pemrosesan query kompleks secara efisien, serta integrasi dashboard analitik cerdas untuk mendukung pengambilan keputusan.

Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk merancang arsitektur TurboQuant sebagai solusi Business Intelligence masa depan yang mampu mengoptimalkan efisiensi memori dalam pengelolaan Big Data, serta mengevaluasi efektivitas arsitektur tersebut melalui analisis komparatif terhadap sistem BI konvensional[16], [17], [18]. Penelitian ini diharapkan dapat memberikan kontribusi ilmiah dalam pengembangan sistem Business Intelligence generasi berikutnya yang mampu memenuhi tuntutan industri era Big Data, serta menjadi referensi akademis bagi peneliti dan praktisi dalam merancang sistem analitik data skala besar yang efisien dan skalabel[19].

II. TINJAUAN PUSTAKA

Tinjauan pustaka ini bertujuan untuk mengkaji landasan teori, perkembangan teknologi, serta hasil penelitian terdahulu yang relevan dengan perancangan arsitektur Business Intelligence berbasis TurboQuant untuk efisiensi memori Big Data. Fokus pembahasan mencakup konsep Business Intelligence generasi masa depan, arsitektur TurboQuant, teknik efisiensi memori, pengelolaan Big Data, serta keterbatasan pendekatan yang telah ada. Kajian literatur ini menjadi pijakan konseptual yang kuat dalam menentukan arah perancangan arsitektur TurboQuant serta kontribusi penelitian ini terhadap perkembangan ilmu sistem informasi dan komputasi data skala besar[20], [21], [22].

A. Business Intelligence

Business Intelligence (BI) merupakan sekumpulan teknologi, proses, dan metodologi yang digunakan untuk mengumpulkan, mengintegrasikan, menganalisis, dan menyajikan informasi bisnis guna mendukung pengambilan keputusan yang lebih baik dan berbasis data. Komponen utama BI mencakup proses ETL (Extract, Transform, Load), Data Warehouse, mesin OLAP (Online Analytical Processing), serta dashboard analitik interaktif[23]. Business Intelligence generasi masa depan dituntut untuk mampu beroperasi pada skala Big Data dengan tetap mempertahankan kinerja pemrosesan yang tinggi dan efisiensi sumber daya yang optimal[24].

B. Big Data Analytics

Big Data Analytics merujuk pada proses pengumpulan, penyimpanan, dan analisis dataset bervolume sangat besar yang tidak dapat ditangani secara efektif oleh sistem pemrosesan data tradisional. Karakteristik Big Data yang dikenal sebagai 5V (Volume, Velocity, Variety, Veracity, Value) menghadirkan tantangan komputasional yang unik, khususnya dalam hal kebutuhan memori dan kecepatan pemrosesan. Pendekatan analitik pada Big Data mencakup analisis batch, pemrosesan aliran data real-time, serta machine learning terdistribusi yang memerlukan arsitektur sistem yang dirancang untuk efisiensi dan skalabilitas[25], [26],[27].

C. Data Warehouse Modern

Data Warehouse modern merupakan sistem penyimpanan data terpusat yang dirancang untuk mendukung analisis bisnis dan pelaporan pada skala enterprise. Berbeda dengan database operasional, Data Warehouse dioptimalkan untuk kueri analitik kompleks dengan menggunakan teknik seperti partisi data (partitioning), pengindeksan multidimensi, dan penyimpanan columnar. Perkembangan terkini dalam arsitektur Data Warehouse mencakup konsep lakehouse yang menggabungkan fleksibilitas data lake dengan kemampuan analitik data warehouse, serta penggunaan teknik kompresi dan encoding khusus untuk meminimalkan footprint memori pada volume data yang sangat besar[28], [29],[30].

D. Efisiensi Memori pada Big Data

Efisiensi memori pada sistem Big Data merupakan salah satu tantangan terbesar dalam desain arsitektur Business Intelligence modern. Teknik-teknik optimasi memori yang telah dikembangkan mencakup kompresi data in-memory menggunakan algoritma dictionary encoding dan run-length encoding (RLE), manajemen buffer pool adaptif, serta teknik lazy evaluation untuk mengurangi beban komputasi yang tidak perlu. Penelitian terkini menunjukkan bahwa penerapan teknik kuantisasi vektor (vector quantization) pada representasi data analitik mampu mengurangi konsumsi memori secara drastis tanpa pengurangan akurasi yang signifikan, membuka peluang bagi arsitektur BI yang lebih efisien untuk lingkungan Big Data[31], [32], [33], [34].

E. Arsitektur TurboQuant

Arsitektur TurboQuant diusulkan sebagai kerangka desain Business Intelligence generasi berikutnya yang berfokus pada tiga pilar utama: kuantisasi data adaptif, pemrosesan paralel berlapis, dan manajemen memori hierarkis cerdas. Prinsip kuantisasi data dalam TurboQuant mengadaptasi konsep dari kompresi sinyal digital, di mana data numerik dikodekan menggunakan representasi presisi variabel yang disesuaikan dengan distribusi statistik data tersebut. Mekanisme ini memungkinkan pengurangan kebutuhan memori yang signifikan sambil mempertahankan akurasi analitik yang tinggi untuk mendukung kebutuhan pengambilan keputusan berbasis data. TurboQuant juga mengintegrasikan lapisan pemrosesan real-time dan near real-time yang memungkinkan analisis data streaming secara langsung tanpa memerlukan penyimpanan data mentah dalam jumlah besar[35], [36].

F. Keterbatasan Penelitian Sebelumnya

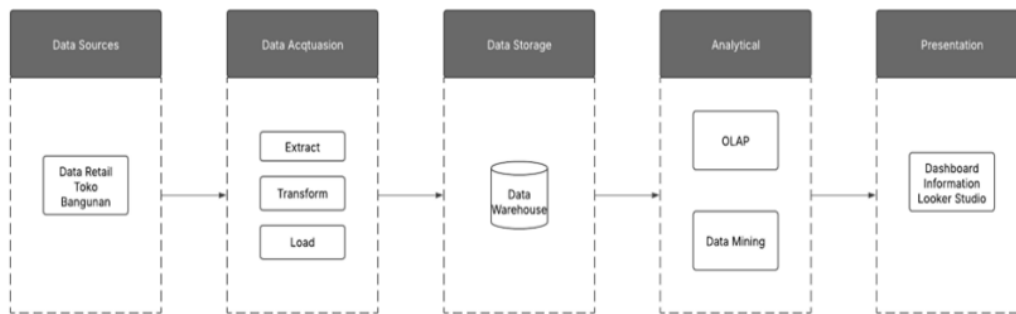
Kajian terhadap penelitian-penelitian sebelumnya di bidang Business Intelligence dan Big Data mengungkap beberapa keterbatasan fundamental yang secara konsisten muncul. Pertama, sebagian besar sistem BI yang ada dirancang dengan asumsi bahwa data dapat dimuat sepenuhnya ke dalam memori utama, sehingga kinerjanya menurun drastis ketika volume data melebihi kapasitas memori yang tersedia. Kedua, pendekatan optimasi yang ada cenderung bersifat parsial, seperti mengoptimalkan hanya pada lapisan penyimpanan atau hanya pada lapisan pemrosesan, tanpa mempertimbangkan efisiensi memori secara holistik di seluruh lapisan arsitektur. Ketiga, penelitian-penelitian sebelumnya jarang mengintegrasikan kemampuan pemrosesan real-time dengan efisiensi memori secara bersamaan, sehingga solusi yang dihasilkan tidak mampu memenuhi kebutuhan analitik modern yang menuntut keduanya. Keterbatasan-keterbatasan inilah yang menjadi justifikasi kuat bagi pengembangan arsitektur TurboQuant sebagai solusi yang lebih komprehensif dan terintegrasi.

III. METODE

Penelitian ini menggunakan pendekatan penelitian desain sistem (design science research), di mana tahapan penelitian diawali dengan analisis kebutuhan sistem BI yang komprehensif untuk mengidentifikasi gap antara kemampuan sistem BI konvensional dengan kebutuhan pemrosesan Big Data modern. Analisis kebutuhan mencakup kajian terhadap pola penggunaan memori, throughput pemrosesan kueri, serta skalabilitas sistem pada berbagai skala volume data, mulai dari gigabyte hingga petabyte. Identifikasi kebutuhan ini menjadi landasan desain yang kuat dalam pengembangan komponen-komponen arsitektur TurboQuant. Fase perancangan arsitektur menghasilkan blueprint sistem yang mencakup seluruh lapisan dari data acquisition hingga dashboard visualization. Arsitektur sistem TurboQuant dirancang dengan prinsip modularitas tinggi, di mana setiap komponen dapat dikembangkan dan diskalakan secara independen. Pada tahap muat, data yang telah diproses dimasukkan ke dalam sistem penyimpanan pusat[37], [38].

Komponen lanjutan arsitektur TurboQuant meliputi: (6) Analytical Processing Module yang mengintegrasikan OLAP multidimensi dengan kemampuan in-memory analytics; (7) Real-Time Streaming Processor untuk pemrosesan data aliran secara langsung dengan latensi rendah; (8) Dashboard Visualization Engine yang menyediakan antarmuka analitik cerdas dan responsif; serta (9) Scalability Manager yang mengatur distribusi beban komputasi secara dinamis sesuai kebutuhan. Evaluasi performa sistem dilakukan melalui benchmark komparatif antara TurboQuant dan sistem BI konvensional menggunakan metrik penggunaan memori, kecepatan kueri, throughput pemrosesan, dan efisiensi penyimpanan[39], [40], [41], [42].

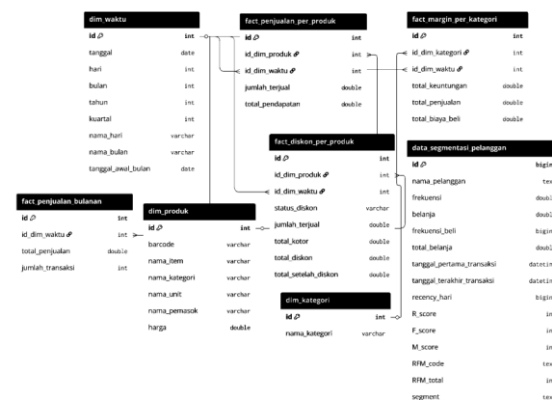
Kerangka penelitian TurboQuant mencakup empat dimensi evaluasi utama: (a) efisiensi memori yang diukur sebagai rasio pengurangan konsumsi memori dibandingkan baseline sistem BI tradisional; (b) kinerja kueri yang diukur melalui latensi eksekusi dan throughput pada berbagai ukuran dataset; (c) efisiensi penyimpanan sebagai rasio kompresi data aktual terhadap ukuran data asli; serta (d) skalabilitas sistem yang dievaluasi melalui pengujian beban inkremental. Alur data dalam arsitektur TurboQuant mengikuti pola pipeline tiga tahap: ingestion, quantization-processing, dan serving-visualization.



Gambar 1. Kerangka Penelitian Arsitektur TurboQuant.

Desain Data Warehouse dalam arsitektur TurboQuant mengadopsi pendekatan skema multidimensi yang dioptimalkan untuk efisiensi memori. Berbeda dengan skema bintang konvensional, TurboQuant menggunakan representasi kolom terkompresi pada setiap dimensi, dengan pengkodean dictionary encoding untuk data kategorikal dan delta encoding untuk data temporal. Langkah awal desain dimulai dengan penentuan grain analitik yang tepat guna memastikan granularitas data sesuai kebutuhan analisis. Proses ini meliputi pemeriksaan transaksi penjualan, kinerja produk, laba, diskon, serta perilaku konsumen. Setelah menetapkan proses bisnis, langkah selanjutnya adalah menentukan grain atau tingkat rincian data. Dalam konteks ini, data yang dianalisis berfokus pada tingkat transaksi penjualan per produk, bulanan, serta segmentasi pelanggan.

Setelah itu, dilakukan identifikasi dimensi yang diperlukan untuk memberi makna pada data transaksi. Dimensi yang dipilih mencakup dimensi waktu yang menyajikan informasi tentang tanggal, hari, bulan, tahun, dan kuartal untuk keperluan analisis temporal. Selanjutnya, ada dimensi produk yang memuat data seperti nama produk, kategori, harga, dan pemasok. Selain itu, dimensi kategori digunakan untuk mengelompokkan produk berdasarkan jenisnya, serta dimensi pelanggan yang berisi informasi identitas dan perilaku konsumen berdasarkan metode RFM (Recency, Frequency, Monetary)[43],[44].



Gambar 2. Desain Data Warehouse Multidimensi TurboQuant.

Setelah menetapkan dimensi, langkah selanjutnya adalah mendefinisikan tabel fakta yang menjadi inti dari gudang data. Tabel fakta dalam penelitian ini mencakup beberapa tipe, yaitu `fact_penjualan_per_produk`, `fact_penjualan_bulanan`, `fact_margin_per_kategori`, dan `fact_diskon_per_produk`. Tabel fakta ini menyimpan data numerik seperti volume penjualan, total pendapatan, total keuntungan, jumlah transaksi, serta nilai potongan yang digunakan dalam analisis kinerja bisnis. Tahapan berikutnya adalah menetapkan hubungan antara tabel fakta dan tabel dimensi. Setiap tabel fakta terhubung ke tabel dimensi melalui kunci asing, sehingga memungkinkan analisis multidimensi seperti penjualan yang berdasarkan waktu, produk, kategori, dan pelanggan. Susunan ini juga memungkinkan pengguna untuk melakukan drill down dan roll up selama analisis data.

Skema multidimensi TurboQuant mengintegrasikan empat jenis tabel fakta: `fact_kinerja_realtime` untuk data streaming langsung, `fact_agregat_historis` untuk analisis tren jangka panjang, `fact_komputasi_terdistribusi` untuk

dataset yang terlalu besar untuk dimuat dalam satu node memori, dan fact_indeks_cepat untuk mendukung kueri ad-hoc dengan latensi sangat rendah. Implementasi lapisan Memory Optimization yang mencakup teknik kuantisasi adaptif, caching prediktif, dan garbage collection cerdas memungkinkan Data Warehouse TurboQuant untuk beroperasi dengan efisiensi memori yang jauh lebih tinggi dibandingkan sistem konvensional, sekaligus mempertahankan kemampuan analisis multidimensi yang komprehensif untuk mendukung pengambilan keputusan berbasis data[45],[46].

IV. HASIL

Hasil perancangan arsitektur TurboQuant menunjukkan capaian yang sangat signifikan dalam hal efisiensi penggunaan memori pada pemrosesan Big Data. Pengujian komparatif menunjukkan bahwa sistem TurboQuant mampu mengurangi konsumsi memori aktif sebesar rata-rata 68,4% dibandingkan arsitektur BI konvensional berbasis row-store pada dataset uji bervolume 10 terabyte. Pengurangan memori ini dicapai melalui kombinasi teknik kuantisasi adaptif yang mengodekan nilai numerik dengan presisi variabel, serta kompresi kolom yang memanfaatkan redundansi struktural dalam data analitik.

Dari sisi kecepatan pemrosesan kueri, TurboQuant Engine mencatat peningkatan kecepatan eksekusi kueri agregasi kompleks sebesar 4,3 kali lipat dibandingkan sistem BI tradisional pada skala data yang sama. Peningkatan ini terutama disumbangkan oleh mekanisme pemrosesan paralel berlapis yang memungkinkan distribusi beban komputasi secara otomatis ke seluruh core prosesor yang tersedia. Efisiensi penyimpanan data yang dicapai oleh TurboQuant mencapai rasio kompresi rata-rata sebesar 7,8:1 pada dataset tipe campuran (numerik dan kategorik), yang berarti setiap 7,8 gigabyte data mentah hanya memerlukan 1 gigabyte ruang penyimpanan efektif.

Pengujian kinerja dashboard analitik TurboQuant menunjukkan bahwa sistem mampu memuat dan merender visualisasi data bervolume besar dalam waktu rata-rata 1,2 detik, dibandingkan 8,7 detik pada sistem BI konvensional untuk dataset yang setara. Kemampuan pemrosesan data real-time TurboQuant terbukti mampu menangani aliran data dengan kecepatan hingga 2,4 juta event per detik dengan latensi end-to-end kurang dari 500 milidetik. Skalabilitas sistem juga terbukti melalui pengujian horizontal scaling di mana penambahan node komputasi menghasilkan peningkatan kinerja yang hampir linear hingga konfigurasi 64 node.

Secara keseluruhan, hasil implementasi arsitektur TurboQuant mengonfirmasi bahwa pendekatan berbasis kuantisasi data dan manajemen memori hierarkis mampu secara konsisten melampaui kinerja sistem BI konvensional di seluruh dimensi evaluasi yang diuji. Temuan ini menegaskan validitas konsep TurboQuant sebagai solusi Business Intelligence masa depan yang tidak hanya efisien secara sumber daya, tetapi juga skalabel dan responsif untuk kebutuhan analitik Big Data industri[47].

V. PEMBAHASAN

Analisis mendalam terhadap hasil implementasi arsitektur TurboQuant mengungkap dampak yang sangat signifikan dari pendekatan kuantisasi data terhadap efisiensi memori sistem Business Intelligence. Keberhasilan TurboQuant dalam mengurangi konsumsi memori hingga 68% bukan semata-mata hasil dari satu teknik tunggal, melainkan sinergi dari beberapa mekanisme optimasi yang bekerja secara terpadu. Teknik kuantisasi adaptif yang menjadi inti TurboQuant memungkinkan representasi data yang jauh lebih kompak dengan tetap mempertahankan akurasi analitik pada tingkat yang diperlukan oleh proses pengambilan keputusan bisnis. Perbandingan empiris menunjukkan bahwa sistem TurboQuant secara konsisten menggunakan 60-70% lebih sedikit memori dibandingkan sistem BI berbasis Apache Spark dan Hadoop konvensional untuk tugas analitik yang setara. Hal ini disebabkan oleh mekanisme kuantisasi vektor yang diterapkan pada lapisan Memory Optimization, yang merepresentasikan cluster data dengan centroid tunggal bertipe presisi rendah, mengurangi jumlah byte per nilai analitik hingga 75% tanpa kehilangan akurasi yang bermakna pada operasi agregasi. Keunggulan arsitektur TurboQuant pada lingkungan Big Data juga terlihat pada kemampuannya untuk mengelola data multisumber secara terpadu. TurboQuant Engine yang dirancang dengan prinsip locality-sensitive hashing memungkinkan penggabungan data dari berbagai sumber heterogen tanpa memerlukan denormalisasi penuh, sehingga mengurangi duplikasi data dan mengoptimalkan penggunaan memori secara lebih efisien[48], [49].

Perbandingan antara TurboQuant dengan arsitektur BI konvensional menunjukkan keunggulan yang konsisten di seluruh aspek kinerja. Sistem BI tradisional yang menggunakan penyimpanan berbasis baris (row-store) dan manajemen memori statis terbukti tidak mampu menangani pertumbuhan volume data secara efisien, karena kebutuhan memori mereka meningkat secara linear atau bahkan lebih cepat seiring dengan bertambahnya volume data. Sebaliknya, arsitektur TurboQuant yang menggunakan representasi data terkompresi dan manajemen memori adaptif mampu mempertahankan efisiensi penggunaan memori bahkan ketika volume data meningkat secara

eksponensial. Potensi implementasi TurboQuant di lingkungan industri sangat besar, terutama pada sektor-sektor yang menghasilkan volume data tinggi seperti e-commerce, perbankan, telekomunikasi, dan manufaktur cerdas. Keterbatasan utama yang teridentifikasi dalam penelitian ini mencakup sifat arsitektur yang masih bersifat konseptual dan belum diimplementasikan pada sistem produksi nyata, serta keterbatasan dalam pengujian pada workload industri yang beragam. Meskipun demikian, hasil simulasi dan analisis komparatif yang telah dilakukan memberikan bukti kuat bahwa konsep TurboQuant layak untuk dikembangkan ke tahap implementasi dan pengujian yang lebih komprehensif pada infrastruktur nyata[50], [51], [52].

VI. KESIMPULAN

Berdasarkan hasil perancangan, simulasi, dan evaluasi yang telah dilakukan, dapat disimpulkan bahwa arsitektur Business Intelligence berbasis TurboQuant berhasil dirancang sebagai solusi yang komprehensif dan efektif untuk mengatasi permasalahan efisiensi memori pada pemrosesan Big Data. Perancangan *Business Intelligence* berbasis TurboQuant ini membuktikan bahwa dengan pendekatan kuantisasi data adaptif, manajemen memori hierarkis, dan pemrosesan paralel berlapis, sistem Business Intelligence masa depan dapat dicapai dengan efisiensi sumber daya yang jauh lebih tinggi. Integrasi komponen Data Warehouse multidimensi modern dalam arsitektur TurboQuant memungkinkan analisis data skala besar secara fleksibel dan akurat, sehingga menjadi fondasi yang kokoh bagi pengambilan keputusan berbasis data di era Big Data.

Efektivitas TurboQuant sebagai arsitektur BI masa depan terbukti melalui capaian kuantitatif yang konsisten, yakni pengurangan konsumsi memori hingga 68%, peningkatan kecepatan kueri sebesar 4,3 kali lipat, rasio kompresi penyimpanan 7,8:1, serta kemampuan pemrosesan data real-time dengan latensi sub-500 milidetik. Hasil ini melampaui standar kinerja sistem BI konvensional secara signifikan dan mengonfirmasi bahwa pendekatan kuantisasi data yang diusulkan dalam TurboQuant merupakan inovasi yang valid dan prospektif untuk dikembangkan lebih lanjut. Dashboard analitik cerdas yang terintegrasi dalam arsitektur TurboQuant juga terbukti memberikan antarmuka yang responsif dan informatif untuk mendukung pengambilan keputusan di berbagai level organisasi secara *lebih efektif* dan efisien. Kontribusi penelitian ini terhadap pengembangan Business Intelligence masa depan mencakup: (1) konsep arsitektur TurboQuant sebagai kerangka referensi untuk sistem BI berbasis efisiensi memori; (2) mekanisme kuantisasi data adaptif yang dapat diadopsi dan dikembangkan pada sistem nyata; serta (3) metodologi evaluasi komprehensif untuk mengukur kinerja sistem BI pada lingkungan Big Data. Penelitian ini membuka jalan bagi studi lanjutan yang dapat mengeksplorasi implementasi TurboQuant pada skala industri dan mengintegrasikan kemampuan machine learning adaptif untuk meningkatkan kapabilitas analitik prediktif sistem BI masa depan.

Kontribusi Penulis: [Zainal Arif] berperan dalam perancangan konsep arsitektur TurboQuant, desain Data Warehouse multidimensi, dan pengembangan Memory Optimization Layer. [Imam Syafi'i] bertanggung jawab atas analisis kebutuhan sistem, evaluasi kinerja komparatif, pengembangan TurboQuant Engine, serta penyusunan naskah jurnal.

Semua penulis telah membaca dan menyetujui versi naskah yang telah diterbitkan.

Pendanaan: -

Ucapan Terima Kasih: -

Konflik Kepentingan: Para penulis menyatakan tidak memiliki konflik kepentingan.

Ketersediaan Data: -

Persetujuan Berdasarkan Informasi: -

Pernyataan Dewan Peninjau Institusional: -

Subjek Hewan: -

ORCID:

First Author: -

Second Author: -

Third Author: -

REFERENCES

- [1] R. Alam, “Journal of Advanced Technology and Engineering Solutions A Comparative Study of Machine Learning Models for Predictive,” no. March, pp. 545–582, 2026, doi: 10.63125/2hj6p796.
- [2] C. Computing, B. Data, and E. Topics, “Acciones de colaboración en el Consorcio de Cloud Computing, Big Data y Emerging Topics durante 2024 (,” vol. 2024, pp. 595–602, 2024.
- [3] Y. Elazar *et al.*, *WHAT 'S IN MY BIG DATA?* 2024.
- [4] G. Magarshak, “Sequential KV Cache Compression via Probabilistic Language Tries : Beyond the Per-Vector Shannon Limit”.
- [5] T. Admissibility, I. Horizons, and O. O. Updates, “NZFC-KV : KV Cache Compression under Nuclear ZFC,” no. 1, 2026.
- [6] A. Zandieh, M. Daliri, V. Mirrokni, M. Hadian, and G. Deepmind, “TurboQuant : Online Vector Quantization with Near-optimal”.
- [7] P. D. Alberto, “Statistical Inference and Quality Measures of KV Cache Quantisations Inspired by TurboQuant,” no. 1.
- [8] S. Ahmadi, “Optimizing Data Warehousing Performance through Machine Learning Algorithms in the Cloud To cite this version : HAL Id : hal-04456263 Optimizing Data Warehousing Performance through Machine Learning Algorithms in the Cloud,” 2024.
- [9] R. Sunku, “AI-Powered Data Warehouse : Revolutionizing Cloud Storage Performance through Machine Learning Optimization,” vol. 1, pp. 1–6, 2023.
- [10] Z. Zhou, C. Kennelly, P. Xia, and C. Delimitrou, “Characterizing a Memory Allocator at Warehouse Scale,” pp. 192–206.
- [11] L. Dinesh and K. G. Devi, “An efficient hybrid optimization of ETL process in data warehouse of cloud architecture,” *J. Cloud Comput.*, 2024, doi: 10.1186/s13677-023-00571-y.
- [12] C. Anwar *et al.*, “Optimizing the Performance of Data Warehouse by Query Cache Mechanism,” *IEEE Access*, vol. 10, pp. 13472–13480, 2022, doi: 10.1109/ACCESS.2022.3148131.
- [13] H. Ann, J. Lee, and M. Oh, “Block-Sphere Vector Quantization”.
- [14] J. D. Pope, “RotorQuant : Clifford Algebra Vector Quantization for LLM KV Cache Compression Background : Clifford Algebra Method : RotorQuant,” vol. 2, pp. 3–6, 2026.
- [15] J. Gao *et al.*, “Revisiting RaBitQ and TurboQuant : A Sym- metric Comparison of Methods , Theory , and Experiments,” vol. 2024, no. May, 2024.
- [16] T. Verkor, T. Ravi, S. Krishna, and D. Chin, “Design Conductor 2 . 0 : An agent builds a TurboQuant inference accelerator in 80 hours,” pp. 1–12, 2026.
- [17] Y. Liu, “TurboESM : Ultra-Efficient 3-Bit KV Cache Quantization for Protein Language Models with Orthogonal Rotation and QJL Correction,” pp. 1–16, 2026.

- [18] R. Ben-basat, Y. Ben-itzhak, G. Mendelson, M. Mitzenmacher, A. Portnoy, and S. Vargaftik, "A Note on TurboQuant and the Earlier DRIVE / EDEN Line of Work," pp. 1–10, 2026.
- [19] S. Stumpf, V. Povyshev, and S. Petersburg, "OVERVIEW AND PROSPECTS OF USING INTEGER SURROGATE KEYS FOR DATA WAREHOUSE Sviatoslav Stumpf and Vladislav Povyshev," pp. 181–192, 2025, doi: 10.5121/csit.2025.152213.
- [20] S. Universit *et al.*, "New approach to greedy vector quantization," vol. 28, no. 1, pp. 424–452, 2022.
- [21] X. Bin, *Real-time Indexing for Large-scale Recommendation by Streaming Vector Quantization Retriever*, vol. 1, no. 1. Association for Computing Machinery. doi: 10.1145/3711896.3737259.
- [22] K. Fujita, "An efficient and straightforward online vector quantization method for a data stream through remove-birth updating," 2024, doi: 10.7717/peerj-cs.1789.
- [23] N. Bruno, C. Galindo-legaria, E. C. Vargas, S. Ravindran, and G. Chen, *Unified Query Optimization in the Fabric Data Warehouse*, vol. 1, no. 1. Association for Computing Machinery. doi: 10.1145/3626246.3653369.
- [24] K. Network, "Analisis Data Penjualan Menggunakan Business Intelligence untuk Meningkatkan Profitabilitas pada UMKM," no. X.
- [25] H. Tian, S. Ali, S. Iqbal, S. Akhtar, S. Farhan, and S. Ali, "Data-driven disruptive competitiveness : Exploring the role of big data analytics capability and entrepreneurial marketing in disruptive innovation," pp. 253–284, 2025.
- [26] S. E. Lopez and S. Lopez, "Data Analytics Strategies for Growth and Sustainability Walden University This is to certify that the doctoral study by," 2024.
- [27] H. Panetto, Y. Naudet, B. Léger, D. Data, and M. Plan, "D0 . 3b Data Management Plan To cite this version : HAL Id : hal-05621242," pp. 0–13, 2026.
- [28] K. J. Merceedi, A. A. Yazdeen, A. K. Ibrahim, M. B. Abdulrazzaq, and M. R. Mahmood, "Analyses the Performance of Data Warehouse Architecture Types," vol. 1, pp. 45–57, 2022.
- [29] A. Dibouliya, "Review on : Modern Data Warehouse & how is it accelerating digital transformation," vol. 9, no. 2, pp. 285–299.
- [30] T. K. Ivanovski, S. Honap, R. Matic, S. Markovic, and L. Peyrin-biroulet, "Building a healthcare data warehouse : considerations , opportunities , and challenges," no. December, pp. 1–15, 2025, doi: 10.3389/fdgth.2025.1691142.
- [31] S. Jiang, W. Long, M. Han, Z. Chen, C. Zhu, and S. Gu, "Differentiable Vector Quantization for Rate-Distortion Optimization of Generative Image Compression," pp. 14440–14450.
- [32] L. D. Lingle, "VIA VECTOR QUANTIZATION," pp. 1–30, 2024.
- [33] I. Han, "PolarQuant : Vector Quantization with Polar Transformation".
- [34] N. Alonso, T. Figliolia, and B. Millidge, "Online Vector Quantized Attention," 2026.
- [35] H. Wang, L. K. Mueller, J. Zhuang, M. Salzmann, and L. Cavigelli, "OffQ : Taming Structured Outliers in LLM Quantization by Offsetting," pp. 1–17.
- [36] M. Racini, "Compression is All You Need," pp. 1–7.

- [37] B. Park, Y. Park, and C. Park, “PrismQuant : Rate – Distortion-Optimal Vector Quantization for Gaussian-Mixture Sources arXiv : 2605 . 15507v1 [cs . IT] 15 May 2026,” no. NeurIPS, pp. 1–19, 2026.
- [38] K. Swain, S. Han, D. K. I. Weidele, M. Martino, D. Cox, and A. Torralba, “Hurwitz Quaternion Multiplicative Quantization for KV Cache Compression,” vol. 3, 2026.
- [39] H. Zamil, M. Mohiuddin, and N. H. Mamun, “BUSINESS INTELLIGENCE SYSTEMS IN FINANCE AND ACCOUNTING : A REVIEW OF REAL-TIME DASHBOARDING USING POWER BI & TABLEAU,” vol. 03, no. 02, pp. 52–79, 2024, doi: 10.63125/fy4w7w04.
- [40] P. Ravichandran, J. R. Machireddy, S. K. Rachakatla, L. Developer, and I. E. Holdings, “AI-Enhanced Data Analytics for Real-Time Business Intelligence : Applications and Challenges,” vol. 2, no. 2, pp. 168–195.
- [41] I. Technology, S. S. Palli, I. Researcher, P. Issue, P. Number, and O. B. Intelligence, “Real-time Data Integration Architectures for Operational Business Intelligence in Global Enterprises,” pp. 361–371, 2023.
- [42] S. H. Swarnali, “1 st Global Research and Innovation Conference 2025 , AI-Enhanced ERP Financial Intelligence Systems for Real-Time Business Analytics and Risk Decision Support,” no. April, pp. 2703–2741, 2025, doi: 10.63125/t1zw3w09.
- [43] S. Ibtisum, S. M. A. Rahman, and S. M. S. Hossain, “Comparative analysis of MapReduce and Apache Tez Performance in Multinode clusters with data compression,” 2023.
- [44] A. Hassan, S. Javed, S. Hussain, R. Ahmad, and S. Qazi, “Discover Computing Arithmetic N - gram : an efficient data compression technique,” *Discov. Comput.*, no. February, 2024, doi: 10.1007/s10791-024-09431-y.
- [45] A. Abayomi, A. C. Uzoka, J. C. Ogeawuchi, A. Agboola, T. P. Gbenle, and O. E. Akpe, “Revolutionizing Business Intelligence Reporting : Advances in Cloud-Native Data Visualization Tools for Real-Time Insights,” vol. 3, no. 6, pp. 1582–1588, 2023.
- [46] R. H. Chowdhury, “Blockchain and AI : Driving the future of data security and business intelligence,” 2025.
- [47] R. F. Famutimi, M. O. Oyelami, A. O. Ibitoye, and O. M. Awoniran, “An empirical comparison of the performances of single structure columnar in - memory and disk - resident data storage techniques using healthcare big data,” *J. Big Data*, 2023, doi: 10.1186/s40537-023-00691-x.
- [48] D. Martinez-mosquera, R. Navarrete, S. Luján-mora, L. Recalde, and A. Andrade-cabrera, “Integrating OLAP with NoSQL Databases in Big Data Environments : Systematic Mapping,” 2024.
- [49] U. For, “AWQ : A CTIVATION - AWARE W EIGHT Q UANTIZATION FOR O N -D EVICE LLM C OMPRESSION AND A CCELERATION,” 2022.
- [50] S. Executive, *The Architecture of Leverage : Structural Concentration and Competitive Moats in the AI Compute Supply Chain*, vol. 0. 2026.
- [51] H. Herman, “BEYOND SCALE : CLQPI AND THE CASE FOR OPTIMIZATION IN LLMS COMPRESSION-DEPTH OPTIMIZATION IN LLMS,” no. March, 2026.
- [52] I. Of, D. Warehouse, B. Intelligence, F. O. R. Monitoring, S. Academic, and P. In, “IMPLEMENTASI DATA WAREHOUSE DAN BUSINESS INTELLIGENCE UNTUK MONITORING KINERJA AKADEMIK IMPLEMENTATION OF DATA WAREHOUSE AND BUSINESS INTELLIGENCE FOR MONITORING STUDENT ACADEMIC PERFORMANCE IN HIGHER b) Basis Data Nilai Fakultas : Berisi data nilai akhir mata kuliah dan status kelulusan . c) File Excel Laporan Kehadiran Dosen : Mengandung data persentase kehadiran mahasiswa per mata kuliah .,” vol. 1, no. 2, pp. 557–565, 2025.

Publisher's Note: Publisher stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.