ISSN: 3079-9325

AUBIQ: A Generative AI-Powered Framework for Automating Business Intelligence Requirements in Resource-Constrained Enterprises

Ruolin Qi[1,*]

[1]Johns Hopkins Carey Business School, Johns Hopkins University, Washington.DC 20001, United States

[*]Corresponding Author

Abstract

The rapid evolution of Business Intelligence (BI) systems has necessitated a paradigm shift from static reporting to dynamic, self-service analytics. However, this transition presents significant challenges for Small and Medium-sized Enterprises (SMEs), where the paucity of technical expertise and the absence of dedicated data engineering teams create a "translation gap" between business intent and technical implementation. This paper introduces AUBIQ (Automated BI Query framework), a novel generative AI-powered system designed to automate the requirements engineering process for BI in resource-constrained environments. By leveraging Large Language Models (LLMs) with a retrieval-augmented generation (RAG) architecture, AUBIQ translates natural language business queries into executable BI specifications and SQL logic without requiring human intervention. This study adopts a Design Science Research methodology to conceptualize, build, and evaluate the framework. The findings indicate that AUBIQ significantly reduces the latency between query formulation and insight generation, achieving a 92.5% accuracy rate in requirement parsing compared to traditional manual methods. Furthermore, the framework demonstrates a marked improvement in user satisfaction metrics among non-technical stakeholders. These results suggest that Generative AI can democratize access to advanced analytics in SMEs, mitigating the dependency on scarce technical resources and enabling more agile decision-making processes.

ISSN: 3079-9325

Keywords

Generative AI, Business Intelligence, Requirements Engineering, SMEs, Natural Language Processing.

1. Introduction

1.1. Research Background

In the contemporary digital economy, data has emerged as a critical asset for organizational survival and competitive advantage. Business Intelligence (BI) systems, which facilitate the transformation of raw data into actionable insights, are no longer a luxury but a necessity for enterprises of all scales. While large corporations have successfully integrated sophisticated BI infrastructures supported by specialized teams of data scientists and engineers, Small and Medium-sized Enterprises (SMEs) face a distinct set of structural hurdles. SMEs are characteristically defined by their resource constraints, specifically in terms of financial capital and specialized human resources. The conventional BI workflow requires a rigorous requirements engineering process where business users articulate their needs, and technical specialists translate these needs into database queries, dashboard schemas, and data pipelines. This dependency on technical intermediaries creates a bottleneck. In the context of SMEs, where a single IT generalist may manage everything from network infrastructure to data analytics, the capacity to rapidly translate analytical needs into technical execution is severely compromised. Consequently, decision-making cycles are elongated, and the potential value of data remains largely untapped.

1.2. Literature Review

The academic discourse surrounding BI requirements engineering (RE) has traditionally focused on structured methodologies such as interviews, workshops, and prototyping. Scholars like Kimball and Ross [1] established the foundational methodologies for dimensional modeling, emphasizing the need for precise requirement definitions to avoid failure in data warehousing projects. However, these traditional methods are labor-intensive and assume the availability of skilled systems analysts, an assumption that often fails in the SME context. With the advent of

ISSN: 3079-9325

Natural Language Processing (NLP), researchers began exploring the potential of Natural Language Interfaces to Databases (NLIDB). Early attempts, as noted by Androutsopoulos et al. [2], were limited by strict syntax rules and domain-specific dictionaries, rendering them brittle in dynamic business environments.

The emergence of Large Language Models (LLMs), driven by the Transformer architecture, has revolutionized this landscape. Recent studies [3] have demonstrated the capability of models like GPT-4 to understand context and generate code. However, existing literature primarily focuses on the technical accuracy of Text-to-SQL generation in isolation. There is a noticeable gap in research regarding holistic frameworks that integrate these generative capabilities into the end-to-end BI requirements lifecycle specifically tailored for the resource constraints of SMEs. Most current frameworks do not adequately address the "hallucination" problems of LLMs when dealing with specific, private enterprise schemas, nor do they provide a robust validation loop for non-technical users.

1.3. Problem Statement

The core problem addressed in this research is the inefficiency and inaccuracy inherent in the BI requirements engineering process within SMEs. Specifically, there exists a semantic gap between the natural language used by business stakeholders to express analytical needs and the structured query languages (SQL) or schema definitions required by BI tools. In resource-constrained enterprises lacking dedicated data analysts, this gap leads to two detrimental outcomes: first, the "time-to-insight" is prohibitively long as non-technical users struggle to formulate queries; second, the potential for error in manual translation leads to mistrust in data outputs. Existing automated solutions are often too expensive or complex for SMEs, leaving a critical need for a lightweight, automated framework that utilizes Generative AI to bridge this gap effectively.

1.4. Research Objectives and Significance

The primary objective of this study is to design, develop, and evaluate AUBIQ, a framework that leverages Generative AI to automate the translation of high-level business goals into precise BI requirements and executable queries. Specifically, this research aims to validate whether a retrieval-augmented generation approach can

ISSN: 3079-9325

improve the accuracy of requirement mapping compared to standard keyword-based methods. The significance of this research lies in its potential to democratize data analytics. By reducing the technical barrier to entry, AUBIQ empowers SMEs to utilize data with the same agility as larger corporations. Theoretically, this study contributes to the literature on AI-augmented Information Systems by providing empirical evidence on the efficacy of LLMs in the specific domain of requirements engineering.

1.5. Thesis Structure

The remainder of this thesis is structured to provide a logical progression from theory to empirical validation. Chapter 2 outlines the research design and methodology, detailing the architectural construction of AUBIQ and the experimental setup used for evaluation. Chapter 3 presents the core analysis and discussion, offering detailed statistical breakdowns of the framework's performance, supported by code implementations and comparative tables. Finally, Chapter 4 and 5 conclude the study by summarizing key findings, acknowledging limitations, and proposing directions for future research.

2. Research Design and Methodology

2.1. General Introduction to Research Methodology

To address the research objectives effectively, this study adopts the Design Science Research (DSR) methodology. DSR is particularly appropriate for Information Systems research where the goal is to create and evaluate a novel artifact—in this case, the AUBIQ framework—to solve a identified practical problem. The research process follows the cycle of problem identification, objective definition, design and development, demonstration, and evaluation. This methodological choice allows for the rigorous construction of the system while ensuring its relevance to the specific constraints of SMEs. The study integrates both qualitative design principles and quantitative evaluation metrics to ensure a comprehensive assessment of the framework's utility.

ISSN: 3079-9325

2.2. Research Framework

The proposed AUBIQ framework operates on a multi-layered architecture designed to ensure accuracy and context-awareness. The process begins with the *Input Layer*, where business users submit natural language queries. This input is processed by the *Contextual Processing Layer*, which utilizes a vector database to retrieve relevant metadata about the organization's data schema (tables, columns, and relationships) without exposing the actual data rows, thereby preserving privacy. The *Generative Layer* then employs a Large Language Model to synthesize the user intent with the retrieved schema to generate specific BI requirements (SQL queries or visualization specifications). Finally, the *Validation Layer* executes a syntax check and presents a simplified explanation back to the user for confirmation. This feedback loop is crucial for mitigating AI hallucinations.

2.3. Research Questions and Hypotheses

This study is guided by two central research questions. First, to what extent can a Generative AI-powered framework accurately translate natural language business queries into executable BI requirements in an SME context? Second, how does the implementation of such a framework impact the operational efficiency of the data analysis process compared to traditional human-mediated workflows? Based on these questions, the study posits the following hypotheses. Hypothesis 1 (H1) suggests that the AUBIQ framework will achieve a requirement parsing accuracy of greater than 85%, significantly outperforming traditional keyword-based NLP approaches. Hypothesis 2 (H2) proposes that the time required to generate actionable insights will be reduced by at least 50% when using the AUBIQ framework compared to manual SQL formulation.

2.4. Data Collection Methods

Data for the evaluation phase was collected through a controlled experiment involving 50 participants from five distinct SMEs operating in the retail and logistics sectors.

ISSN: 3079-9325

These sectors were chosen due to their heavy reliance on timely data for inventory and supply chain management. The participants were divided into two groups: a control group using standard SQL interface tools and an experimental group using the AUBIQ interface. A dataset of 200 standardized business queries, ranging from simple aggregations (e.g., "Total sales last month") to complex multi-join logic (e.g., "Identify customers who bought Product X but not Product Y in Q3"), was constructed to serve as the testing benchmark. Ground truth answers were established by senior data architects to serve as the reference point for accuracy calculations.

2.5. Data Analysis Techniques

The analysis of the collected data utilizes quantitative statistical methods. To evaluate the accuracy of the generated requirements, precision, recall, and F1-scores are calculated by comparing the AUBIQ-generated SQL against the ground truth queries. Execution efficiency is measured by recording the timestamp from query submission to successful result generation. Comparative analysis between the control and experimental groups is conducted using independent t-tests to determine statistical significance. Additionally, a post-experiment survey based on the System Usability Scale (SUS) is employed to quantify user satisfaction and perceived ease of use. The qualitative feedback from the survey is analyzed using thematic analysis to identify recurring pain points or benefits.

3. Analysis and Discussion

3.1. Implementation and Code Analysis

The core of the AUBIQ framework is built upon a Python-based backend that orchestrates the interaction between the user interface, the vector store for schema context, and the LLM. The implementation focuses on prompt engineering strategies that enforce strict adherence to the provided database schema, minimizing the risk of

ISSN: 3079-9325

the model inventing non-existent table names. The following code segments illustrate the critical components of the system.

Code Snippet 1: Context Retrieval and Prompt Construction

```
import openai
from vector_db import SchemaStore

class RequirementGenerator:
    def __init__(self, api_key, schema_store):
        self.client = openai.OpenAI(api_key=api_key)
        self.schema_store = schema_store

    def generate_bi_requirement(self, user_query):
        # Retrieve relevant table schemas based on user query embedding
        relevant_context = self.schema_store.search(user_query, top_k=3)

        # Construct the system prompt with strict constraints
        system_prompt = f"""
        You are an expert BI Analyst for an SME.
        Translate the user's natural language question into a precise SQL
query.

        Use ONLY the following schema schema definitions:
        {relevant_context}
```

ISSN: 3079-9325

```
        Rules:
        1. Do not use tables or columns not listed above.
        2. If the logic requires a JOIN, ensure keys match the schema.
        3. Output the SQL inside a JSON object with keys: 'sql',
'explanation'.
        """

        response = self.client.chat.completions.create(
            model="gpt-4-turbo",
            messages=[
                {"role": "system", "content": system_prompt},
                {"role": "user", "content": user_query}
            ],
            response_format={"type": "json_object"}
        )

        return response.choices[0].message.content
```

The code above demonstrates the Retrieval-Augmented Generation (RAG) pattern. By dynamically injecting only the relevant parts of the database schema into the system_prompt, the model's focus is narrowed, significantly reducing the computational overhead and the probability of error. The embedding search ensures that even if the database has hundreds of tables, the LLM only receives the context for the specific tables relevant to the query (e.g., "Sales" and "Customers"), preventing the context window from being exceeded.

ISSN: 3079-9325

Code Snippet 2: Automated Syntax Validation and Error Handling

```
import json
import sqlalchemy
from sqlalchemy.exc import SQLAlchemyError

def validate_and_execute(db_engine, llm_response):
    try:
        response_data = json.loads(llm_response)
        generated_sql = response_data['sql']
        explanation = response_data['explanation']

        # Dry-run execution to check for syntax errors without committing
        with db_engine.connect() as connection:
            # Using begin() starts a transaction that rolls back automatically
if needed
            # For SELECT statements, we just want to ensure it runs
            result_proxy =
connection.execute(sqlalchemy.text(generated_sql))
            columns = result_proxy.keys()

        return {
            "status": "success",
            "columns": list(columns),
            "sql": generated_sql,
```

ISSN: 3079-9325

```
        "explanation": explanation
    }

except SQLAlchemyError as e:
    return {
        "status": "error",
        "error_type": "Database Execution Error",
        "message": str(e.__dict__['orig'])
    }
except json.JSONDecodeError:
    return {
        "status": "error",
        "error_type": "Parsing Error",
        "message": "Model output was not valid JSON."
    }
```

This validation logic serves as a critical guardrail. Before any result is presented to the business user, the system attempts a dry-run of the query. If the database engine returns an error (e.g., ambiguous column name), the system captures this error. In a more advanced iteration of AUBIQ, this error message is fed back into the LLM for self-correction, creating an autonomous healing loop. This programmatic validation ensures that syntactically incorrect queries are never presented to the user, thereby building trust in the system's reliability.

ISSN: 3079-9325

3.2. Quantitative Performance Analysis

The experimental results provide compelling evidence supporting the efficacy of the AUBIQ framework. The performance was evaluated across three dimensions: Query Success Rate, Logical Accuracy, and Time Efficiency. The descriptive statistics of the participant demographics and the dataset complexity are presented below.

Table 1: Descriptive Statistics of Experimental Dataset and Participants

Metric Category	*Variable*	*Frequency / Mean*	*Percentage / SD*
Participants	*Total Participants*	*50*	*100%*
	Non-Technical Business Users	*35*	*70%*
	Technical / IT Staff (SME)	*15*	*30%*
Query Complexity	*Simple (Single Table Aggregation)*	*80*	*40%*
	Moderate (1-2 Joins, Filters)	*80*	*40%*
	Complex (Subqueries, Window Functions)	*40*	*20%*
Data Context	*Total Tables in*	*45*	*-*

ISSN: 3079-9325

	Schema		
	Average per Table	Rows150,000	± 50,000

As shown in Table 1, the distribution of query complexity was balanced to simulate real-world usage scenarios. The majority of participants were non-technical, which aligns with the target demographic of the study. The system's performance against these queries was compared with a baseline approach (standard SQL generation without RAG context) and manual writing by the IT staff participants.

Table 2: Comparative Analysis of Accuracy Metrics

Method	*Precision*	*Recall*	*F1-Score*	*Execution Success Rate*
Baseline (Standard LLM)	*0.68*	*0.72*	*0.70*	*65.0%*
Manual (IT Staff)	*0.94*	*0.91*	*0.92*	*96.0%*
AUBIQ Framework	*0.91*	*0.93*	*0.92*	*92.5%*

The data in Table 2 reveals that the AUBIQ framework significantly outperforms the standard LLM baseline. The F1-Score of 0.92 indicates that AUBIQ achieves parity with human experts (Manual IT Staff) in terms of logical accuracy. The standard LLM often failed because it hallucinated table names that did not exist in the SME's specific database. By injecting the schema context, AUBIQ corrected these

ISSN: 3079-9325

hallucinations. Although the manual approach by IT staff still holds a slight edge in execution success rate, the gap is marginal (3.5%).

3.3. Discussion of Operational Efficiency

While accuracy is paramount, the primary value proposition for SMEs is efficiency and speed. The analysis of time expenditure highlights the transformative potential of automation.

Table 3: Time Efficiency Analysis (Time in Minutes)

Query Complexity	*Manual Formulation (Mean Time)*	*AUBIQ Generation (Mean Time)*	*Improvement Factor*
Simple	*5.4 min*	*0.2 min*	*27x*
Moderate	*12.8 min*	*0.5 min*	*25.6x*
Complex	*35.2 min*	*1.1 min*	*32x*
Average	*17.8 min*	*0.6 min*	*29.6x*

Table 3 illustrates a dramatic reduction in time-to-insight. For complex queries, where human users typically spend significant time understanding schema relationships and debugging syntax errors, AUBIQ reduced the process to just over a minute. This massive improvement factor (approximately 30x) validates H2, suggesting that the bottleneck in SME analytics is not the data processing itself, but the requirement translation phase.

Table 4: Error Categorization Analysis

Error Type	*Description*	*Frequency (Baseline)*	*Frequency (AUBIQ)*	*Reduction*
Hallucination	*Inventing*	*45*	*4*	*91%*

ISSN: 3079-9325

	non-existent columns/tables			
Syntax Error	Invalid SQL syntax	12	2	83%
Logic Mismatch	Correct syntax, but wrong intent	13	9	30%
Ambiguity	Model asks for clarification	0	15	N/A

Table 4 provides a granular look at failure modes. The most significant finding is the 91% reduction in Hallucination errors. The AUBIQ framework's architecture, which constrains the LLM to the retrieved schema, effectively grounds the model. Interestingly, the "Ambiguity" category increased for AUBIQ; this is a positive design feature where the system was programmed to ask clarifying questions rather than guessing, further ensuring reliability.

4. Conclusion and Future Directions

While the AUBIQ framework has demonstrated robust performance in Retrieval-Augmented Generation (RAG) based Text-to-SQL tasks, current capabilities represent only a preliminary stage in the automation of Business Intelligence (BI). To fully surmount the data barriers faced by Small and Medium-sized Enterprises (SMEs), future research must transcend simple "query translation" and evolve towards architectures that are more autonomous, inclusive, and adaptive. This

ISSN: 3079-9325

necessitates a focus on Agentic Workflows, the integration of unstructured data, and the establishment of continuous learning mechanisms.

4.1. Transitioning from Passive Generation to Active Agency

The current system operates primarily as a linear, one-shot generator, a mechanism often insufficient given that real-world data environments are characterized by ambiguity, inconsistent schema naming, and noise, frequently leading to logic errors in single-pass generation. This limitation stems from the inherent lack of real-time environmental awareness and the inability to execute dynamic, modular actions. The central, groundbreaking trajectory for future research lies in the development of Agentic Workflows, often leveraging structured frameworks like ReAct (Reasoning and Action) and Hierarchical Reinforcement Learning (HRL) to manage the decomposition of complex, multi-stage BI objectives. Here, the Large Language Model (LLM) functions not merely as a passive coder, but as an autonomous "intelligent agent" capable of advanced planning, execution monitoring, and self-correction through dynamic tool-calling architectures.

To achieve this paradigm shift, agents must possess the capacity for Autonomous Exploration and Verification. Rather than relying solely on abstract schema metadata, the agent should perform active data profiling and schema introspection—a "pre-query" process—to interrogate the database and its current state before formulating the final SQL. For instance, upon receiving a query regarding "sales in the East Region" and profitability analysis, the agent must execute a series of exploratory queries, utilizing Goal Decomposition to break the analysis into verifiable steps. This process, an implementation of the Chain of Thought (CoT) reasoning, fundamentally mimics a human analyst's initial data inspection and is crucial for mitigating logic mismatch errors derived from value hallucination or stale metadata. Furthermore, the development of robust internal planning modules must include mechanisms for Ethical Constraining — ensuring the agent's planned queries do not access restricted datasets or violate internal data usage policies, effectively building compliance directly into the Model-Based Planning (MBP) structure.

Furthermore, the robustness of the system hinges on its ability to incorporate Iterative Debugging and Self-Correction. When the database returns an

ISSN: 3079-9325

error, the agent should intelligently rewrite the query based on reflective reasoning. This establishes a structured feedback loop that is vital for handling complex multi-step analysis. However, researchers must also rigorously address the Cost Optimization imperative; this increased autonomy mandates a necessary trade-off analysis balancing execution latency, the volume of API calls, and the marginal gain in accuracy. The long-term scalability of this architecture requires exploring Federated Learning or Transfer Learning strategies, allowing the agentic models deployed across a non-competitive consortia of SMEs to learn from a collective pool of successful query execution paths, thereby overcoming the inherent "small data" challenge that limits individual SME training datasets. The aim is a refined, optimized agent model that achieves maximum reliability with minimal, predictable latency, securing the economic viability of the zero-intervention user experience.

4.2. Transcending Structural Boundaries: Multimodal Data Lakehouse Integration

The confinement of current research to structured relational databases (SQL) is increasingly restrictive, overlooking the vast proportion of business insights—often representing early warning signals or qualitative context—locked away in unstructured data silos. Therefore, future research must resolutely expand from "Text-to-SQL" to "Text-to-Insight," achieving a seamless, unified fusion of structured and unstructured retrieval within a coherent analytical framework.

This requires the development of sophisticated Hybrid Retrieval Architectures that integrate high-performance Vector Databases with traditional SQL query engines within a modern Data Lakehousecontext. Research must focus on achieving Zero-Shot/Few-Shot Schema Linking in this hybrid environment, enabling the LLM to link abstract concepts in unstructured text to specific columns in the SQL schema. This capability is paramount for tackling Temporal Reasoning in multimodal queries, allowing the AI to accurately link time-series sales performance (structured) with the precise unstructured events (e.g., a specific marketing email or service outage record) that occurred during that period.

Achieving this fusion inevitably introduces the profound Semantic Alignment Challenge and the need for rigorous Cross-Modal Grounding. Advanced machine

ISSN: 3079-9325

learning mechanisms are required to map subjective, qualitative textual descriptions (e.g., "poor build quality") to specific, quantitative indicators in the relational data. Solving this demands substantial contributions from Ontology Engineering, utilizing controlled vocabularies and hierarchical business models to give the AI the formal structure needed to reliably reconcile disparate data formats. The resulting system must also support Dynamic Knowledge Graphs, where relationships are constantly updated based on new data and user interactions. Furthermore, the Economic Viability of building such an infrastructure for SMEs must be proven through scalable, cloud-native deployments. Given the sensitivity of unstructured data, strict Data Governance protocols must be paramount, focusing on automated Personal Identifiable Information (PII) masking and compliance with complex regulations concerning Data Sovereignty and cross-border data transfer, making the system legally sound and ethically responsible for globalizing SMEs. This final technical evolution requires the system to function not just as an information retrieval tool, but as a robust, domain-specific reasoning engine.

5. Sociotechnical Implications and Organizational Transformation: Reshaping Data Literacy

The introduction of AUBIQ is not merely a passive technical upgrade but a powerful catalyst for profound organizational restructuring and cultural evolution. This final chapter examines the long-term sociotechnical implications of such automation tools, focusing on the inevitable evolution of employee skills, the ethical and psychological risks of algorithmic dependency, and the deep cultural shifts required for successful, sustainable adoption, culminating in policy recommendations.

5.1. The Semiotic Shift in Data Literacy: From Syntax to Semantics

The advent of Natural Language Interfaces (LUI) effectively breaks the historical barrier of technical proficiency, moving the cognitive burden away from syntax (how to write the code) toward semantics (how to formulate the question). This liberation does not diminish the need for Data Literacy; rather, it elevates and redirects its definition. The cognitive burden is definitively shifted to epistemic responsibility: employees must now possess heightened rigorous logical thinking to clearly define

ISSN: 3079-9325

business metrics boundaries and understand the implications of their queries. This fundamental change catalyzes a Value Chain Transformation, where the cost of data labor approaches zero, shifting organizational value upstream to strategy and downstream to action. This introduces a new domain of AI-Mediated Communication, where employees must not only communicate their needs but also understand the meta-communication regarding the AI's limitations and its potential for reflecting biases present in its RAG sources—a key AI Ethicsconcern that necessitates formal bias detection routines within the retrieval pipeline.

Consequently, Prompt Engineering will rapidly emerge as a fundamental vocational skill, acting as a new form of Requirements Engineering. Staff must be trained to translate vague, qualitative business intuition into precise, unambiguous logical instructions for the AI. This mandates that SMEs restructure their training programs to incorporate Metacognitive Training alongside Prompt Engineering, teaching users how to monitor their own understanding and when to doubt the AI's output. Furthermore, a formal Framework for Responsible Prompting (FRP) is essential to ensure ethical data requests and guard against the unintentional introduction of bias. The ultimate objective is the establishment of Algorithmic Accountability, where the organization clearly defines who — the user, the model developer, or the data governance team—is responsible for a materially incorrect query result.

5.2. Algorithmic Dependency and the Risk of De-skilling

While automation offers critical efficiency gains, organizations must remain vigilant regarding the severe potential for De-skilling within the workforce. This risk is often driven by the Black-Box Trust Paradox: human operators are psychologically prone to automation bias and automation complacency, leading to a dangerous erosion of their critical assessment skills. This vulnerability directly impacts risk perception and can lead to systemic decision errors, creating the philosophical challenge of Epistemic Substitution, where human expertise is entirely replaced by, and dependent upon, algorithmic output.

To mitigate this dangerous trajectory, future research must formalize governance frameworks that mandate strict Human-in-the-Loop (HITL) protocols, coupled with sophisticated Adaptive Trust Modeling—the system's ability to learn when the human

ISSN: 3079-9325

user is reliable or overloaded, and when the human should rely on the AI. This requires the establishment of organizational structures—such as mandatory code review cycles for high-impact queries—and the integration of enhanced Explainability (XAI) mechanisms. XAI must offer Post-Hoc Explanations detailing why the LLM chose specific schema elements and filtering conditions, thereby providing a comprehensible audit trail. This mechanism is vital for maintaining a critical, reflective lens within the organization, fostering accountability, and fulfilling requirements related to Regulatory Compliance (such as the need to provide a "Right to Explanation" under data protection mandates).

5.3. Organizational Agility and Cultural Restructuring

For SMEs, adopting GenAI-driven BI tools is fundamentally a matter of cultural change that enables true Data Democratization. The technology decentralizes decision authority by empowering frontline staff to access self-service data insights directly, promoting a flattening of the organizational hierarchy and enhancing overall organizational agility. This transformation requires significant investment in Change Management to systematically manage the workforce transition and ensure organizational buy-in.

Furthermore, the framework compels a profound cultural shift from experience-driven to rigorous evidence-based management. This transformation is key to building Organizational Resilience, which must be formally measured not just by efficiency gains, but by the measurable improvement in decision quality and the speed of market adaptation. This necessity drives the development of new Organizational Measurement techniques to accurately capture the true ROI of AI adoption. Ultimately, the successful SMEs of the future will be those that effectively translate the computational power of AI into an organizational cognitive surplus. To ensure equitable access and accelerated adoption, this research concludes with a strong policy recommendation for state and national bodies to provide comprehensive technological and financial support, recognizing that the proliferation of advanced BI tools in SMEs, facilitated by Global Standardization bodies (e.g., ISO, OECD) for ethical and technical interoperability, is a strategic necessity for enhancing National AI Strategy competitiveness and sustainable economic growth.

ISSN: 3079-9325

References

[1] Kimball, R., & Ross, M. (2013). The data warehouse toolkit: The definitive guide to dimensional modeling (3rd ed.). John Wiley & Sons.

[2] Androutsopoulos, I., Ritchie, G. D., & Thanisch, P. (1995). Natural language interfaces to databases – An introduction. Natural Language Engineering, 1(1), 29–81.

[3] OpenAI. (2023). GPT-4 technical report. arXiv preprint arXiv:2303.08774.

[4] Wixom, B. H., & Watson, H. J. (2001). An empirical investigation of the factors affecting data warehousing success. MIS Quarterly, 25(1), 17–41.

[5] Baars, H., & Kemper, H. G. (2008). Management support with structured and unstructured data—An integrated business intelligence framework. Information Systems Management, 25(2), 132–148.

[6] Lewis, J. R. (2018). The System Usability Scale: Past, present, and future. International Journal of Human–Computer Interaction, 34(7), 577–590.

[7] Chaudhuri, S., Dayal, U., & Narasayya, V. (2011). An overview of business intelligence technology. Communications of the ACM, 54(8), 88–98.

[8] Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. MIS Quarterly, 28(1), 75–105.

[9] Popescu, A. F., Etzioni, O., & Kautz, H. (2003). Towards a theory of natural language interfaces to databases. In Proceedings of the 8th International Conference on Intelligent User Interfaces (pp. 149–157). ACM.

[10] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30, 5998–6008.

[11] Chen, H., Chiang, R. H. L., & Storey, V. C. (2012). Business intelligence and analytics: From big data to big impact. MIS Quarterly, 36(4), 1165–1188.

[12] Tou, Y., Cheng, J., & Davison, R. M. (2020). Genesis of the giant: A case study of the evolution of the data capability. International Journal of Information Management, 54, 102146.

[13] Glavic, B., Miller, R. J., & Alonso, G. (2020). The future of data integration: A survey. IEEE Data Engineering Bulletin, 43(1), 12–25.

ISSN: 3079-9325

[14] Brynjolfsson, E., & McAfee, A. (2014). The second machine age: Work, progress, and prosperity in a time of brilliant technologies. W. W. Norton & Company.