movate®

Cracking the Text-to-SQL Challenge:

How to Build Accurate, Production-Ready Query Generation AI Chatbots

Table of Contents

01. Why Text-to-SQL Suddenly Matters More Than Ever

As businesses continue to innovate and democratize data access, Text-to-SQL capabilities have quickly become one of the most desired features across today's analytics stack.

This technology provides a path toward a world where anyone, not just data engineers can simply ask questions using plain English and receive reliable insight (in real-time) from their complex databases.

However, in real production environments, this dream quickly collides with a harsh reality: **accuracy issues**. Models that perform flawlessly in demos often falter when exposed to messy schemas, vague user queries, and the unpredictable nature of enterprise data.

So how do you turn a Text-to-SQL chatbot into a reliable, scalable, production-ready system? Here are the strategies that move the needle.

02. The Real Reason Production Accuracy Is Hard

Deploying Text-to-SQL in production is challenging because real databases and real user behavior introduce complexities that LLMs don't naturally understand.

Here's why accuracy often drops outside controlled environments:

1. Production databases often carry highly complex, enterprise-scale schemas that differ significantly from simplified development setups.
2. Table and column names are sometimes vague, misleading, or non-descriptive, making intent hard to interpret.
3. Large portions of tables, fields, and values come with little to no documentation or metadata to guide understanding.
4. Real-world user queries frequently require multi-table joins, increasing the difficulty of generating accurate SQL.
5. Some tables include hundreds of poorly defined columns, adding ambiguity and noise for the LLM.
6. The LLM may receive an incomplete, missing, or incorrect business context, leading to incorrect query generation.
7. LLMs lack an inherent grasp of absolute or relative time concepts like "yesterday," "last week," or "current month."
8. User questions are often ambiguous, unclear, or poorly phrased, making intent extraction challenging.
9. Many user questions reference implicit data filters on specific columns that the LLM has no visibility into.

03. How to Achieve Higher Accuracy & Consistency in SQL-Generating LLM Systems

Achieving reliable, production-grade SQL generation requires a combination of the right model, clean data foundations, enriched metadata, strong business context, semantics layer, and robust validation of workflows. **Here's a consolidated view of the essential practices:**

1 LLM Model Selection

LLMs such as Azure OpenAI/OpenAI GPT-4o, GPT-5, and Gemini 1.5 Pro are all good models for SQL code generation, and any of these can be used. For data security reasons, we suggest using the Azure OpenAI GPT-4o sandbox environment.

For the chatbot, we were using approximately

15,000

total tokens on average,

so GPT-4o and GPT-5 were well suited for this. However, when we experimented with the cheaper GPT-5 model, we observed higher latency in query generation. Using an open-source model such as Llama-4 can also be considered since it has a strong foundation and was trained on SQL code datasets.

2 Database Schema & Data Analysis

A deep understanding of the underlying schema drives better prompts and more accurate SQL outputs. Teams should map tables, columns, relationships, and explore data types and low-cardinality dimensions (gender, country code, status, department, etc.). This helps models understand how data is structured and where joins are expected.

3 Curated Dataset

When databases suffer from poor quality or involve heavy data transformations, it's often better to build a clean data mart. A clean, simplified dataset becomes the primary query source for the AI system, improving both reliability and speed.

4 Metadata Enrichment & Optimization

Metadata enrichment and optimization involve providing clear definitions for tables and columns in the prompt when names are not well defined and filtering out any tables or columns that are irrelevant to the business use case to reduce noise, maintain a focused context window, and improve accuracy. It also requires clearly defining the available tables, their columns, relationships, and column values in the prompt, so the chatbot can generate accurate SQL.

5 Relevant & Minimal Business Context

Relevant and minimum business context should be included in the prompt to reduce noise and keep the model focused. Prescience Biz Genie provides a dedicated space to define both general operational guidelines and specific business rules that the LLM must follow during query generation, such as date and time formats or valid values for each column. To ensure the LLM always interprets queries accurately based on the present moment, the application should dynamically generate the current date and time and inject it directly into the prompt.

6 Prompt Engineering

Prompt Versioning: It allows teams to track token usage, evaluate how different prompt versions perform, and ultimately deploy the most effective and efficient prompt.

A/B Testing: It helps in identifying the better prompt for performance and optimal cost of use in production.

Custom Prompt: Biz Genie includes a default system prompt, but administrators may need to customize it to improve accuracy. This feature allows admin to override and refine the default prompt.

7 Automated Query Validation

Include clear, non-negotiable instructions in your main system prompt to the LLM to only return SELECT queries and not to include dangerous operation like DROP, DELETE, UPDATE etc.

To avoid SQL injections, make sure Query is validated before it is executed and data is retrieved.

Once query is generated, use **Abstract Syntax Tree** (**AST**) to check these dangerous operations in the generated query.

If generated query syntax is incorrect, then retry mechanism can be included with the last query validation error message, which helps LLM to avoid same error in the next query.

8 Handling Natural Language Ambiguity

When a user's query is unclear or incomplete, the LLM is instructed to ask a clarifying question to the user rather than attempting to provide an incorrect answer.

9 Automated Testing in CI/CD

LLMs are non-deterministic and after every prompt or business context change, testing all user questions can be time consuming and painful, so it is important to automate LLM prompt automation testing. Opensource test frameworks such as DeepEval, Promptfoo, OpenAI Evals are good to use in CI/CD pipelines.

10 N-Shot Learning Examples

In Biz Genie we designed two categories of examples for N-shot learning: **Core examples** and **Semantic search examples.** Core examples are always included in the prompt. They provide consistent guidance on SQL structure, JOIN patterns, and overall query formatting. Semantic search examples are retrieved dynamically.

Only the most relevant examples based on the user's questions are included to minimize token usage while maintaining accuracy.

Vector embedding model: For our use case we used OpenAI "text-embedding-3-small" which is good in terms of performance and cost efficient; for best semantic search "text-embedding-3-large" can be considered.

11 Learning from User Feedback

When an end user "Likes" a correct answer, it triggers an admin review workflow. After validation, the question and its corresponding SQL query are added to the model's example set, strengthening its semantic understanding and improving consistency for similar future queries. These feedback-driven examples are also filtered so that only the most relevant question–query pairs are included in the user prompt, ensuring token size and maintaining high accuracy.

12 Use Vector DB Instead of SQL DB

Use Vector DB instead of SQL DB: while vector embeddings can be stored in SQL DB as float ARRAY, for small scale this may work but for large scale vector search performance can take a hit.

For scalability, faster and accurate retrieval, we should use Vector DBs with **ANN Indexes** support such as **Milvus, Pinecone, Weaviate, ChromaDB, pgvector** etc. We selected pgvector since our use case was not very complex.

Even PostgreSQL **pgvector** has performance issues when searching and retrieving from millions of vectors. In such scenario it is better to use a full fledge vector DB such as **Milvus, Pinecone, ChromaDB** etc. Unlike SQL Databases, vector DBs scale horizontally like distributed database systems and able to handle enterprise data volume.

movate®

13 **Fine-Tuning the LLM**

Fine tuning an LLM model can provide better query accuracy, but it should be considered only as a last resort. When all other options have been exhausted and the required accuracy is still not achieved, fine tuning becomes a viable choice. This approach requires a well-prepared training dataset, typically consisting of 400–500 high-quality examples.

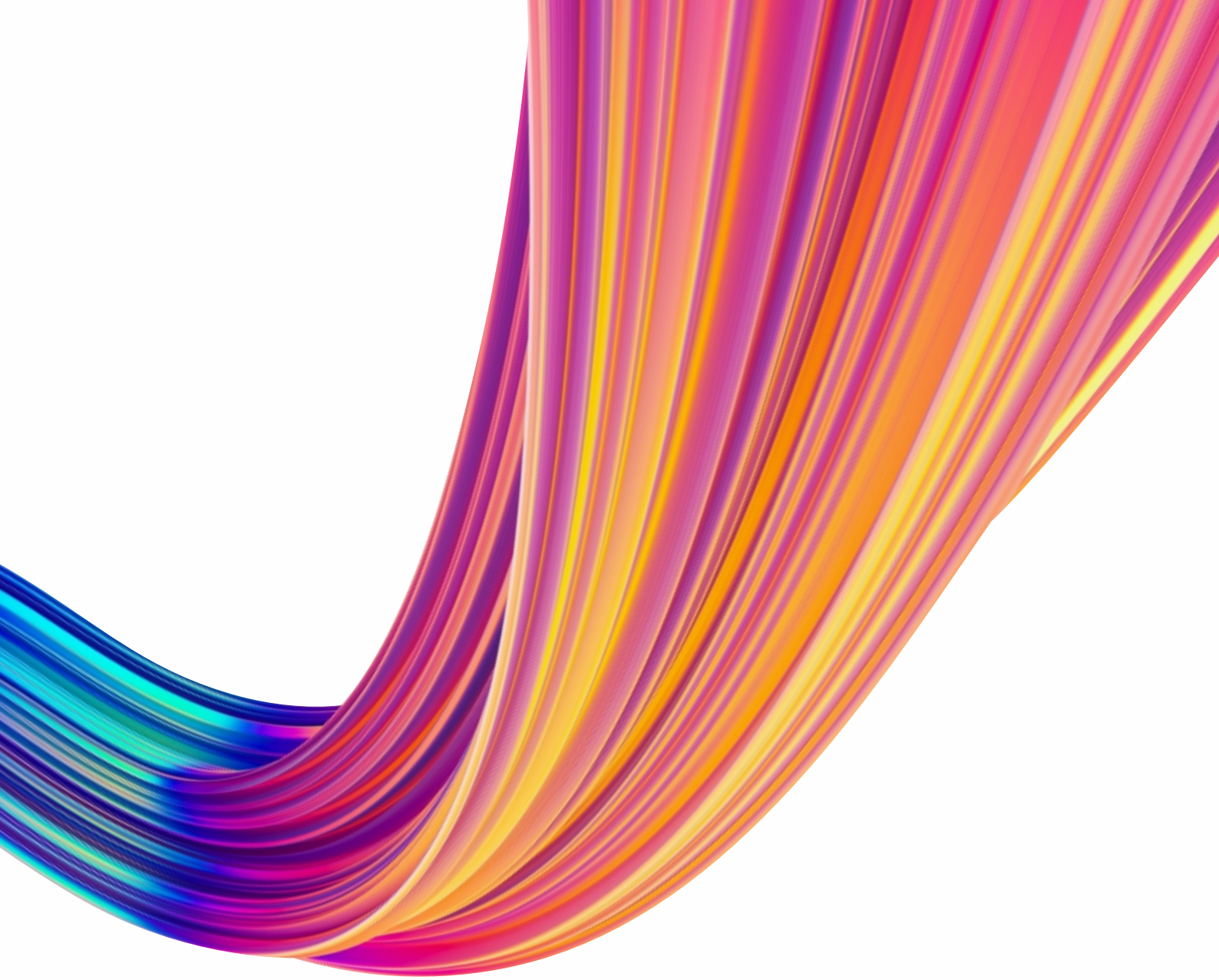
04. Final Thoughts

One key learning is that deploying an AI chatbot isn't a "**set it and forget it**" exercise. Because LLMs are probabilistic, they require ongoing monitoring, refinement, and optimization to maintain high query accuracy. It's equally important to set the right expectations and help customers understand that the process is iterative and evolves over time, it WILL NOT achieve ~99% accuracy on day 1.

In our own deployment, the chatbot started with an accuracy as low as ~40%. By analyzing the customer's data, defining the right business context, adding targeted examples, and continuously incorporating user feedback, we were able to elevate its performance to nearly 90%.

05. About the Author

Umesh Bhatt, AVP at Prescience Decision Solutions, brings over 20 years of experience in the IT industry, specializing in the design and development of enterprise-grade software products and scalable SaaS applications. His core expertise spans system and enterprise storage solutions as well as EdTech platforms, where he has led the creation of robust, high-performance systems that support complex business and learning needs.



About Movate

Movate is a digital technology and customer experience services company committed to disrupting the industry with boundless agility, human-centered innovation, and a relentless focus on driving client outcomes. Recognized as one of the most awarded and analyst-accredited companies in its revenue range, Movate helps ambitious, growth-oriented companies across industries stay ahead of the curve by leveraging its world-class talent of over 12,000+ full-time Movators across 21 global locations and a gig network of thousands of technology experts across 60 countries, speaking over 100 languages.

For more details, please mail us at info@movate.com

www.movate.com

movate®