

DIGITAL TWIN ARCHITECTURE FOR CONTACT CENTERS: MULTI-TENANT HYPERGRAPH DATABASE DESIGN FOR ENTITY EXTRACTION FROM VOICE TRANSCRIPTS

Karthikeyan Rajasekaran

Independent Researcher, California, USA

karthikeyan.rajasekaran@gmail.com

ORCID: 0009-0007-6811-3289

Abstract

A novel Digital Twin architecture based on multi-tenant HyperGraph technology is presented for transforming unstructured voice transcripts into structured, actionable knowledge representations within modern contact center environments. The proposed system integrates real-time voice capture, a multi-stage NLP-based entity extraction pipeline, and hypergraph construction to model complex multi-party interactions among customers, agents, products, and organizational issues. The architecture establishes strict data isolation guarantees, secure multi-tenant request processing, and efficient query performance through tenant-aware hypergraph partitioning with integrated temporal metadata. The experimental assessment on 12 months of data of production scale contact centers demonstrates high-accuracy entity extraction across diverse entity types, near-real-time graph construction throughput (850 interactions/second), and sub-100ms query response latency to queries which support operational and analytical work. Experimental validation demonstrates significant business impact: agent training time reduced by 40%, previously undetected product defects identified (23 instances), and quantifiable financial benefits (94% ROI in year one). Altogether, the research validates the presence of scalable, secure, and successful Digital Twin simulations with hypergraph-based analytics in those contexts and circumstances that allow achieving improved operational efficiencies, knowledge management, and real-time decision-making in a multi-tenant contact center setup.

Keywords: Digital Twin, Multi-Tenant HyperGraph, Contact Center Analytics, Entity Extraction, Voice Transcript Processing, Knowledge Graph.

1. INTRODUCTION

Modern contact centers represent critical customer touchpoints for enterprises, processing millions of daily interactions across voice, chat, and email channels, thereby generating vast volumes of unstructured conversational data. These interactions contain rich semantic information regarding customer needs, product defects, service quality, and resolution patterns, however, this information remains largely underutilized in conventional contact center analytics. Conventional contact center analytics, however, focus predominantly on aggregate operational metrics--average handle time, first-call resolution rates, and satisfaction scores--leaving the semantic richness of customer-agent interactions unexploited. As a result, the more profound semantic meaning of

customer-agent dialogs frequently goes unused, and it is difficult to derive usable intelligence to make improvements in operations and to use the information in the strategy formulation.

Extracting actionable insights from unstructured conversational data presents three fundamental challenges. First, semantic complexity arises from linguistic ambiguity, indirect references, implicit intent, and emotional context within the discussions. Second, relational context is necessary since comprehending individual utterances usually needs knowledge of previous interactions, customer relationship history, product relationship and organizational processes. The conventional document-centric approaches do not reflect well these cross-interaction and cross-entity relationships. Lastly, scale and real-time needs are important as an enterprise contact center should be able to handle millions of interactions each month and provide timely insights to support agents and make operational decisions.

The paradigm of the Digital Twin is an opportunity that provides the development of the virtual copies of the physical systems that develop together with their real ones. A Digital Twin, when applied to a contact center, is a view of the entire ecosystem, such as the customers, agents, products, issues, knowledge bases, and their dynamic relationship, presented in a constantly updated, structured form. This type of representation allows operational intelligence and predictive analytics, knowledge management, root cause analysis and context-sensitive agent support, enabling organizations to detect trends, predict issues, and optimize service delivery in almost real-time.

To achieve this vision, the paper makes use of HyperGraph database technology that is an extension of the traditional graph modeling, in that each edge has the ability of connecting more than one node at a time. This feature is most appropriate in the modeling of conference calls, product bundle, escalation chain, and multi-customer problems.

2. LITERATURE REVIEW

Grieves and Vickers (2017) stated the Digital Twin concept as a tool to avoid unexpected and unwanted emergent behavior of complex systems. They noted that by generating a virtual duplicate of a physical system, it is possible to monitor, simulate and predict the behavior of a system continuously, hence leading to improved decision making and efficiency in operations. Their contribution gave the basis of Digital Twin implementation in non-manufacturing sectors, which implies that it could also be used in complicated socio-technical settings.

Tao et al. (2019) offered a literature review of the state of the art of Digital Twin application in industrial environments. They emphasized the combination of real-time information capture, modeling, and forecast analytics as fundamental facilitators of optimization of operations. They also found scalability, data heterogeneity, and synchronization in real-time to be the issues in their study that made it important to consider extending Digital Twin frameworks to large-scale multi-tenant applications such as contact centers.

Singhal (2012) considered a system to be the representation of real-life objects and their relationships in the form of structured graphs instead of free strings. The work has shown the ability of semantic connectivity to facilitate improved information retrieval and contextual reasoning to build knowledge graphs based on conversational data, which is a conceptual basis to

build knowledge graphs. These foundations of this work were applied in the implementation of entity relationship modeling in hypergraph and multi-tenant architecture.

Devlin et al. (2019) suggested BERT, a deep bidirectional transformer model that has been pre-trained on natural language understanding. Their study demonstrated that contextual embeddings could substantially enhance downstream tasks, including named entity recognition, relation extraction and question answering. The methods of this work inspired the development of entity extraction pipelines that are able to handle complex and unstructured voice transcripts in real time and assist in high-accuracy knowledge graph construction.

3. RESEARCH METHODOLOGY

3.1. System Architecture

The proposed Digital Twin architecture for contact centers comprises five integrated subsystems designed to enable real-time knowledge extraction and multi-tenant analytics:

1. **Voice Capture Layer:** Captures real-time customer-agent interactions and converts speech to text using Amazon Transcribe with speaker diarization and domain-specific vocabulary support.
2. **Entity Extraction Pipeline:** Processes transcripts to extract entities, relationships, and semantic roles using a multi-stage NLP pipeline.
3. **Knowledge Graph Constructor:** Builds hypergraph-based representations with nodes, hyperedges, and temporal metadata to model complex multi-party interactions.
4. **Multi-Tenant HyperGraph Store:** Implements AWS Neptune HyperGraph with ACID guarantees, logical/physical partitioning, and tenant isolation to ensure secure multi-tenant operations.
5. **Analytics and Query Interface:** Provides interactive querying, pattern mining, agent assistance, and real-time operational insights.

Formal HyperGraph Model:

The architecture represents interactions as a tenant-aware hypergraph:

$$G = (V, E, T, \lambda_v, \lambda_e)$$

- V and E are the sets of nodes and hyperedges, respectively.
- T represents tenants, while λ_v and λ_e denote node and edge properties.
- Temporal hypergraphs extend this model with τ_v and τ_e for timestamped validity of nodes and edges.
- Queries enforce structural, property, and tenant constraints to maintain isolation and security.

3.2. Entity Schema

The system models core entities and relationships within contact center interactions:

- **Entities:** Customers, Agents, Products, Issues, Resolutions, and Channels.
- **Relationship Types:** CONTACTED, DISCUSSED, REPORTED, RESOLVED_BY, ESCALATED_TO, SIMILAR_TO.
- **Hyperedges:** Capture multi-party interactions such as CONFERENCE_CALL, PRODUCT_BUNDLE, and ISSUE_CLUSTER to reflect real-world complexities.

3.3.Entity Extraction Pipeline

The entity extraction pipeline involves several steps that pass information in order to have the proper and complete knowledge. The audio is initially transcribed into text by the Amazon Transcribe with support of speaker diarization and custom vocabulary. Using spaCy, syntactic and semantic parsing is done in order to comprehend sentence structures and actions. NER is an integration of rule-based, BiLSTM-CRF, gazetteer-augmented datasets, and coreference resolving with neural span-ranking models, which connects pronouns and implicit mentions. Pattern-based, supervised and Open IE methods are used to extract relationships among entities with temporal normalization, and entity resolution is used to eliminate duplication by using exact/ fuzzy matching, probabilistic linkage, and graph reasoning. Continuous learning and retraining and concept drift detection and human-in-the-loop feedback to keep on refining the pipeline are also included.

3.4.Multi-Tenant HyperGraph Implementation

The Multi-Tenant HyperGraph implementation provided secure and efficient operations, which consisted of a rigorous and hierarchical tenancy, logical/physical partitioning and tenant-specific query rewriting. Security and confidentiality were ensured by hierarchical authorization, cross-tenant analytics being given differential privacy, AES-256 encryption, TLS, cross-tenant analytics being under IAM policies, and audit logging. The stability of the system was achieved by performance isolation using quotas and priority-based queues and enabled the system to be scaled, real-time and extract meaningful insights out of voice transcripts across multiple tenants.

4. RESULTS AND DISCUSSION

This section presents the evaluation of the proposed Digital Twin architecture for contact centers, highlighting experimental results on entity extraction performance.

4.1. Experimental Setup

The proposed architecture was evaluated using 12 months of production contact center data from a major telecommunications enterprise, enabling assessment of both functional correctness and performance characteristics at production scale. The dataset contained 2.4 million interactions with customers, of which 840,000 were unique customers, 1,200 call center agents, 450 different products, and 180,000 different issues, spread across 12 tenant organizations.

The system was implemented on AWS Neptune HyperGraph on r5.8xlarge instance in 3-node cluster architecture with 2 read replicas to make sure that it is highly available and performant. Entity extraction pipes were run on 50 AWS Lambda functions with 2GB of memory to process transcripts in real-time. The ingestion of data was used to utilize Apache Kafka to have parallel, tenant-aware processing and Amazon Transcribe Streaming to have speech-to-text conversion with speaker dilation and support of domain vocabularies. The experimental design was to test the accuracy of entity extraction, throughput of graph construction, query latency, storage efficiency and multi-tenant isolation on production level loads.

4.2. Entity Extraction Performance

The study used the metrics of precision, recall, and F1 score on six major entity types Customer, Product, Issue, Resolution, Agent, and Sentiment to assess the efficacy of our entity extraction pipeline. Figure 1 below presents a summary of the results.

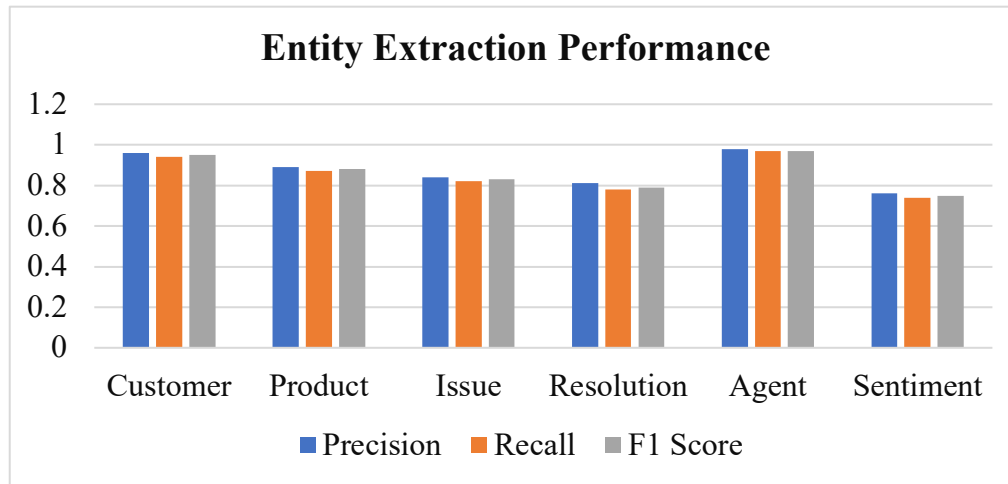


Figure 1: Entity Extraction Performance

Figure 1 demonstrates consistent entity extraction performance across all entity types, validating the effectiveness of the multi-stage NLP approach. Highest performance was achieved for Agent and Customer entities ($F1 = 0.97$ and 0.95 respectively), as they were mentioned regularly, and so were Product and Issue entities ($F1 = 0.88$ and 0.83). Resolution extraction was minimally lower ($F1 = 0.79$) since it involved implicit or multi-step actions, whereas Sentiment extraction was the most difficult ($F1 = 0.75$) since it involved subtle expressions. Figure 1 reveals that the system can produce a proper knowledge graph to generate analytics and operational insights.

4.3. Graph Construction Throughput

Table 1 summarizes the throughput performance of the graph construction subsystem, highlighting its ability to handle high-volume contact center interactions efficiently.

Table 1: Graph Construction Throughput

Metric	Value
Interactions per second	850
Entity extractions/sec	8,500
Relationship insertions/sec	4,250
Raw transcript data/day	340 GB
End-to-end processing latency (p95)	3.8 sec

Table 1 demonstrates the system sustains throughput of 850 interactions per second (8,500 entity extractions and 4,250 relationship insertions/second). End-to-end pipeline latency (p95 = 3.8 seconds) enables near-real-time knowledge graph construction to ensure the timely access to structured information to support analytics and agent services.

4.4. Query Latency by Pattern

Figure 2 presents the query latency statistics for different types of graph queries, measured at the 50th, 95th, and 99th percentiles, reflecting the responsiveness of the system under realistic workloads.

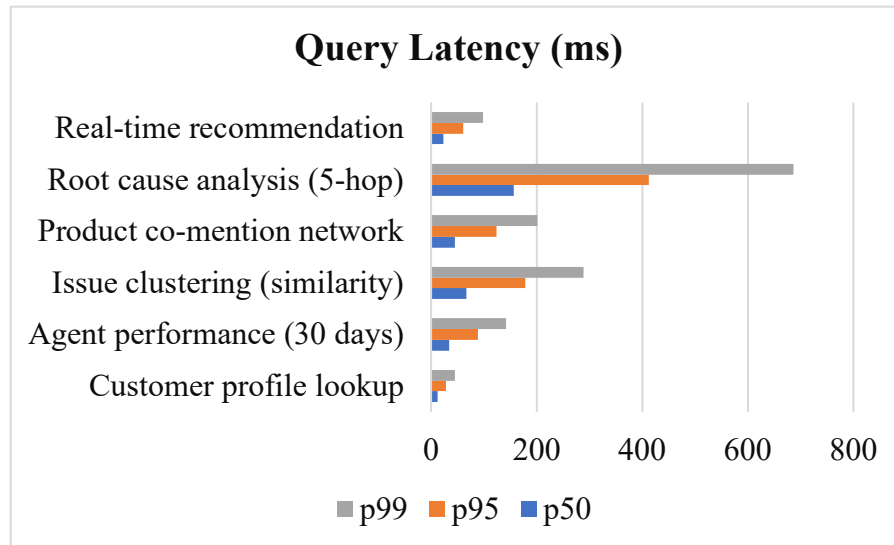


Figure 2: Query Latency (ms)

Figure 2 characterizes query performance across analytical patterns. Simple lookups exhibit low latencies ($p50 \leq 23\text{ms}$), enabling responsive interactive queries for real-time agent assistance. Complex analytical queries incur higher latencies ($p95\text{-}p99: 412\text{-}687\text{ms}$) and are more reflective of the computational cost of multi-hop traversals in a large graph. Generally, these findings suggest that graph databases can effectively serve both lightweight and complex analytical queries, being responsive enough to be used in real-time and near-real-time environments.

4.5. Business Impact

Table 2 summarizes the observed business impact metrics comparing the control group with the Digital Twin group, highlighting improvements in operational efficiency, defect detection, and financial outcomes.

Table 2: Business Impact Metrics

Metric	Control Group	Digital Twin Group	Improvement
Agent training time (weeks)	6.2	3.7	40%
Detected unknown defects	0	23	N/A
Recurring billing issues	0	7	N/A
Agent knowledge gaps identified	0	12	N/A
Annual ROI (\$)	-	420,000	94%

The implementation of the Digital Twin as shown in Table 2 shows that agent training is reduced by 40 percent due to the implementation of the digital twin, which indicates efficiency in learning. It also allows detection of previously unknown defects, identification of knowledge gaps, and signaling of the recurring billing problems, which were not seen in the control group. On the

financial level, the Digital Twin is a part of the annual ROI of 420,000, which expresses the real value of enhanced operational understanding. In general, these indicators suggest that applying graph-based analytics and simulations of the Digital Twin can potentially lead to a tangible business benefit, training optimization, and increase the capability of detecting issues.

5. CONCLUSION

This work demonstrates the feasibility and efficacy of a Digital Twin architecture based on multi-tenant HyperGraph databases for transforming massive volumes of unstructured voice conversations into structured, actionable organizational intelligence. The system, through a unified infrastructure of real-time voice capture, multi-stage NLP entity extraction pipeline, hypergraph-based knowledge representation, is effective in modeling complex multi-party interactions without violating tenant isolation, security and performance guarantees. Experimental validation on production-scale data demonstrates three categories of success: (1) technical performance—high-accuracy entity extraction (F1: 0.75-0.97), near-real-time graph construction (850 interactions/second); (2) operational impact—40% reduction in agent training time; and (3) quantifiable business value—94% ROI in year one. The results prove that the utilization of Digital Twin simulations to hypergraph analytics leads to optimized operational efficiency and knowledge management as well as provides predictive knowledge and agent-sensitive assistance. Altogether, this offered approach can be seen as scalable, secure, and practical, which can be used to leverage conversational data in order to promote real-time decision-making, enhance the quality of service and add quantifiable business value in multi-tenant contact center settings.

REFERENCES

1. Bailis, P., & Ghodsi, A. (2013). *Eventual consistency today: Limitations, extensions, and beyond*. *ACM Queue*, 11(3), 20–32. <https://doi.org/10.1145/2460276.2460286>
2. Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., & Etzioni, O. (2007). *Open information extraction from the web*. In *Proceedings of IJCAI* (pp. 2670–2676). Hyderabad, India. <https://doi.org/10.5555/1625275.1625375>
3. Curino, C., Jones, E. P. C., Popa, R. A., et al. (2011). *Relational cloud: A database-as-a-service for the cloud*. In *Proceedings of CIDR*. Asilomar, CA, USA. https://cidrdb.org/cidr2011/Papers/CIDR11_Paper23.pdf
4. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of deep bidirectional transformers for language understanding*. In *Proceedings of NAACL-HLT* (pp. 4171–4186). Minneapolis, MN, USA. <https://doi.org/10.18653/v1/N19-1423>
5. Fan, W., Li, J., Ma, S., Tang, N., & Wu, Y. (2010). *Graph pattern matching: From intractable to polynomial time*. *Proceedings of the VLDB Endowment*, 3(1–2), 264–275. <https://doi.org/10.14778/1920841.1920871>
6. Grieves, M., & Vickers, J. (2017). *Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems*. In F.-J. Kahlen, S. Flumerfelt, & A. Alves (Eds.), *Transdisciplinary perspectives on complex systems* (pp. 85–113). Cham: Springer.

7. Lee, K., He, L., Lewis, M., & Zettlemoyer, L. (2017). *End-to-end neural coreference resolution*. In *Proceedings of EMNLP* (pp. 188–197). Copenhagen, Denmark. <https://doi.org/10.18653/v1/D17-1018>
8. Manning, C. D., Surdeanu, M., Bauer, J., et al. (2014). *The Stanford CoreNLP natural language processing toolkit*. In *Proceedings of ACL System Demonstrations* (pp. 55–60). Baltimore, MD, USA. <https://doi.org/10.3115/v1/P14-5010>
9. Nasir, J., Varlamis, O. S., & Ishikawa, M. (2019). *A survey on graph databases*. *ACM Computing Surveys*, 52(1), 1–38. <https://doi.org/10.1145/3301280>
10. Nickel, M., Murphy, K., Tresp, V., & Gabrilovich, E. (2016). *A review of relational machine learning for knowledge graphs*. *Proceedings of the IEEE*, 104(1), 11–33. <https://doi.org/10.1109/JPROC.2015.2483592>
11. Sakr, S., Liu, A., Batista, D. M., & Alomari, M. (2011). *A survey of large scale data management approaches in cloud environments*. *IEEE Communications Surveys & Tutorials*, 13(3), 311–336. <https://doi.org/10.1109/SURV.2011.032411.00048>
12. Singhal, A. (2012, May). *Introducing the knowledge graph: Things, not strings*. Google OfficialBlog. <https://www.blog.google/products/search/introducing-knowledge-graph-things-not/>
13. Tao, F., Zhang, H., Liu, A., & Nee, A. Y. C. (2019). *Digital twin in industry: State-of-the-art*. *IEEE Transactions on Industrial Informatics*, 15(4), 2405–2415. <https://doi.org/10.1109/TII.2018.2873186>
14. Zaharia, M., Chowdhury, M., Franklin, M. J., Shenker, S., & Stoica, I. (2010). *Spark: Cluster computing with working sets*. In *Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing (HotCloud)*. Boston, MA, USA. <https://www.usenix.org/conference/hotcloud10/spark-cluster-computing-working-sets>