Towards Leveraging Artificial Intelligence for NoSQL Data Modeling, Querying and Quality Characterization

Chaimae Asaad
Alqualsadi, Rabat IT Center, ENSIAS, Mohammed V University in Rabat
TicLab, Faculty of Engineering and Architecture, International University of Rabat
Morocco
chaimae.asaad@uir.ac.ma
https://orcid.org/0000-0002-3587-1944

***Abstract*—In the last two decades, NoSQL databases emerged and formed an umbrella category grouping well over a hundred databases of different characteristics, and providing a new take on scalability, availability, consistency and data modeling, with the aim of conquering the classic one-size-fits-all solution represented in traditional databases. NoSQL databases' heterogeneity, flexibility and high performance allowed them to encompass the volume, velocity and variety challenges brought by the Big Data era and to fulfill the complex requirements of real-time applications. These advantages have given them a competitive advantage in the database market. However, despite these added bonuses, a wider adoption of NoSQL has been hindered by a few challenges. The "schema-less" nature of NoSQL databases allowing data to be directly ingested, without defining a schema *a priori*, can be a deterrent for many users given its potential impact on data management and integrity, application complexity and system integration. Additionally, the heterogeneous nature of NoSQL databases creates a complex and diverse landscape of query languages and interfaces requiring extensive and wide-ranging expertise. In an effort to breach the standardization and democratization of NoSQL databases, I propose an approach integrating artificial intelligence techniques in NoSQL life cycles, namely: data modeling, querying and quality assessment. This approach offers multi-faceted contributions to the literature and the industry, including: a uniform design methodology for NoSQL databases, an intelligent bi-directional mapping between higher and lower level NoSQL schema representations, the generation of logical and physical NoSQL data models from requirements specified in natural language, intelligent and bi-directional generation of a NoSQL query from a prompt expressed in natural language, and a multi-stakeholder NoSQL database-agnostic quality characterization framework. With this approach, I aim to pave the way towards a fully integrated artificially intelligent system capable of undertaking the processes of data modeling, natural language querying and quality assurance of NoSQL databases. This objective, when achieved, would not only increase the adoption of NoSQL databases by both experts and novices, but would also potentially positively impact the cost and time constraints of NoSQL database handling, as well as their evolvability, maintainability and portability.**

***Index Terms*—NoSQL Databases, Database Design, Query Languages, Data Model Quality Assesment, Artificial Intelligence, Natural Language Processing**

I. INTRODUCTION

The rising pressure to meet industrial demands amid the proliferation of data sources (e.g., social media, Internet of Things) in the Big Data era highlighted the need for flexibility and scalability in storage systems [1]. With the aim of tackling demanding requirements, several databases capable of handling substantial volumes of unstructured and semi-structured data emerged under the umbrella category of NoSQL [1]. "NoSQL" evolved from the Strozzi-coined term referring to a relational database that did not expose a SQL interface like its counterparts, to "Not Only SQL", and now designates a heterogeneous schema-less class of databases with 'non-traditional' technical characteristics and data model. With the creation of data stores by Google, Amazon and several open-source developers, NoSQL technology bloomed, leading to several publicationsn particulary by M. Stonebraker [2], [3], R. Cattell [4], B. Tudorica et al. [5], and J. Han [6]. Eventually, research began evolving in more foundational directions [7], along with new attention paid to the lack of a standard to model NoSQL databases accepted by both the business and academic communities [8].

NoSQL databases are generally categorized into Document, Key-Value, Column-Family and Graph databases [1], [7], [8]. The main strengths of NoSQL databases stem from the flexibility and heterogeneity they offer, the volume of data they can handle, as well as horizontal scalability, high availability, fault-tolerance, schema-less design and eventual consistency [7], [8].

These advantages, however, hinder their interoperability, and require major adjustments in order to operate them in conjunction with each other, whether in the context of polyglot persistence, multimodel databases or in other real-world applications.

And so, this is the main motivation behind this work. The potential of NoSQL to revolutionize the database ecosystem can only be achieved if and when they can be accessible in terms of required expertise, designable and inter- mappable in terms of data modeling, and queryable with ease.

To that end, my work proposes an approach incorporating artificial intelligence techniques to construct a fully integrated system for NoSQL data modeling, querying and quality characterization life cycle. This objective would have a substantial impact on the democratization of NoSQL database use.

The remainder of this paper is structured as follows: In **Section 2**, a detailed problem description is presented, and the research questions targeted in my thesis are dissected. In **Section 3**, related work is presented alongside highlighted gaps in literature. The proposed approach addressing each research question is presented in **Section 4**. Evaluation methods proposed are illustrated in **Section 5**. A conclusion as well as an overview of the current status of my PhD research are presented in **Section 6**.

II. Problem Description

In relational database design, three levels of representation are well established and applied by both academic and business communities: the conceptual, logical, and physical models [8].

The low-level physical model depends on the target database's specifications, and is acquired through the conversion of logical data models. In this context, it includes predeployment design, table definitions, normalization, primary and foreign key relationships, and basic indexing. At a higher representation level, the conceptual model is used to represent a real-world problem. Entity-Relationship (ER) models and Unified Modeling Language (UML) class diagrams are the most widely used representations. The logical model is defined between the conceptual and physical layers, and offers an intermediary representation of how the system should be implemented, regardless of the target database [8].

In NoSQL database design, representation levels are not clear [8], nor are they enforced or widely followed, especially given the schemaless nature of NoSQL databases.

Shin et al. [9] proposed adopting this design process to NoSQL databases, and specified the components relevant to each layer. In a systematic literature review of NoSQL modeling aspects, Vera et al. [8] found no works that followed all three layers, but stated that there is evidence that premodeling NoSQL following the three levels of representation could present benefits.

Vera et al. [8] include entities, attributes, and relationships as components to be defined at the conceptual design phase, in addition to system requirements, number of records, and CRUD operations. The logical design phase includes the category of NoSQL database to be used (e.g., document), as well as target-database-independent features (e.g., embedded/reference relationships). As for the physical level, it includes the selection of a target NoSQL system with adequate data storage features and features.

NoSQL data modeling is still novel. There are academic works proposing ways to model at a physical design level, and some at a logical design level. Other works generally propose a UML or ER model to represent data, to be mapped to a NoSQL schema (*Sec. 3. Related Work*).

The task of unifying major differences in data model characteristics or mapping diverse structures is extremely challenging. To alleviate and elevate this process, I derive the first research question *(RQ1) : How can Artificial Intelligence techniques be leveraged for the unification and mapping of NoSQL data models in all representation levels?*

The diverse landscape of NoSQL databases is reflected in the various query languages used to handle data. Unlike Relational databases which use the SQL standard query language, NoSQL databases do not have a standard query language [10].

Querying is specific to the target database, and often requires a high threshold of specialized knowledge in multiple databases [10], thus creating a hurdle for a great number of potential users who, at the beginning of the appearance of NoSQL, were already deterred by the lack of usage of declarative language SQL [10].

This complexity and lack of accessibility of NoSQL query languages has added a layer of difficulty to the adoption of NoSQL. To this end, I derive the second research question *RQ2: How can Artificial Intelligence be leveraged for the conceptualization of a lingua franca for NoSQL query languages and for the inclusion of natural language querying for NoSQL?*

Because NoSQL design is generally performed at a physical level, especially in practice, the concept of quality is almost never defined in the context of data models. Instead, the literature mainly focuses on the assessment of performance, write/read operations, and other database characteristics of NoSQL. Consequently, there is no characterization of the concept of quality in a database-agnostic context. Given the importance of quality assessment in any life cycle, and particularly for NoSQL where flexibility rules and best practices serve as minimal guidelines for design, I derive the third and last research question of my work *RQ3: How can Artificial Intelligence be leveraged to conceptualize a NoSQL database-agnostic data model quality characterization framework and assist in quality assessment?*

III. Related work

In this section, I present existing work that addresses the research questions and outline the gaps and shortcomings that I target in my research.

Academic works addressing the modeling aspect of NoSQL fall under the contexts of: guidelines, process transformation, query oriented, schema generation, evaluation, ontology, benchmark, and migration [8]. In a systematic literature review, Vera et al. found that most of the works proposed data modeling in the context of guidelines and process transformation. Banerjee et al. [11] proposed an ontology driven metamodel to conceptualize data representation independently of any particular NoSQL database. A common conceptual level abstraction for NoSQL databases is proposed, and a formal vocabulary is implemented using the Protégé tool based on the OWL format to generate logical or physical schemas. Abdelhedi et al. [12] used model automatic transformation in Model Driven Architecture (MDA) to transform a conceptual model and generate NoSQL physical models. A logical model is proposed encompassing aggregate-oriented types of NoSQL. Vera et al. [13] approaches data modeling for NoSQL, specifically for document oriented databases.

SOS (Save Our Systems) and NoAM (NoSQL Abstract Model) were proposed by Atzeni et al. [14], [15] for uniform access and database design of NoSQL databases. SOS is a metamodel designed to represent schemas of aggregate-based store, while NoAM was designed as an intermediate representation to transform aggregate objects of database applications into NoSQL data. Candel et al. [16] devised U-Schema to have a uniform representation able to capture data models of NoSQL and relational data models. This representation supports aggregate-oriented NoSQL databases and graph databases, and outlines mappings between the proposed U-Schema and logical document, key-value, column-family and graph schemas.

In the context of query generation, Candel et al. [17] created SkiQL language designed on U-Schema [16] to achieve a platform-independent schema query service. The Natural Language Tool Kit (NLTK) and BERT (Bidirectional Encoder Representations from Transformers) language model were used for the conversion of natural language queries to non-relational query, specifically MongoDB [18], and a question answering (QA) system was developed to allow users to query external NoSQL databases using natural language [19]. Some works performed SQL query generation from natural language [20], [21]. The literature lacks proposals on how to assess the quality of NoSQL data models, and is instead mainly focused on the issue from a physical-level angle, i.e., studying the performance and physical characteristics of the database itself. For instance, Klein et al. [22], [23] used a quality-attribute method to guide their evaluation of NoSQL databases in the healthcare domain. They evaluated transaction performance, partition-tolerance and data model mapping for three NoSQL databases of different types: MongoDB (Document oriented), Cassandra (Column Family) and Riak (Key Value). Gómez et al. [24] proposed a set of metrics (Existence of types and collections; nesting depth; width of documents; referencing rate; and redundancy) reflecting key aspects of the complexity of document-oriented schemas, with the purpose of facilitating schema analysis and comparison. Many characteristics specific to NoSQL have been studied in the literature. For instance, in an empirical study on the design and evolution of NoSQL database schemas, Scherzinger et al. [25] take into consideration the schema size as well as its denormalized state, since denormalization is a common and recommended practice guideline in NoSQL data modeling.

In summary, there are some approaches addressing parts of the research questions devised in this work, however, to the best of my knowledge, none cover all sub-aspects in combination or incorporate AI, as I intend to do in my work. Further details will be given in *Section 4*.

IV. Proposed Approach

Our approach encompasses three major components, corresponding to the research questions devised in *Section 2.* In what follows, we detail each component. Although each of them represent standalone contributions, the final objective is for them to function in tandem and represent a fully integrated system.

A. RQ1: AI for NoSQL Design

This component includes three phases: (i) the 'translation' of user requirements written in natural language to a unified conceptual graph metamodel, and the subsequent (ii) mapping from this high-level representation to logical metamodels, and (iii) to low-level physical design representations in target databases. These transformations are bi-directional and all require inputs and generate outputs.

(i) Conceptual Modeling Phase: From Natural Language User Requirements to the Graph Unified MetaModel 'NoDeSQL'

In order to transform user requirements expressed in natural language, natural language processing techniques and machine learning techniques are used. After cleaning the text, A Named Entity Recognition model tags each remaining word based on a classification into pre-defined classes. The aim is to recognize the words that reflect "entities" and "structures", as well as the "relationships" or "verbs". Using the results of this process, an intuitive graphical representation is built. This representation uses "circles" to represent "entities" and "lines" to represent "relationships". This intermediary graph represents the business view of the data model, and offers a simple graphical view of the requirements that can be understood by all involved stakeholders. This graph is then mapped to the graph unified data model "NoDeSQL", which is formalized as a property graph with hypergraph extensions. This structure allows for both a higher generalization and flexibility for modeling. "NoDeSQL" is a Graph G comprised of : nodes n, edges e, node label nl, edge label el, additional (optional) hyperedges he.

(ii) Logical Modeling Phase: From NoDeSQL to Logical NoSQL Data Models

At this level, the "NoDeSQL" Graph Unified MetaModel had been formalized. A logical representation version "NoDeSQLog" is generated, including Node Properties Np, Edge Properties Ep, and HyperEdge Properties Hep. This graph must be complete, i.e., it must contain all the information from the requirements and the conceptual model. Mappings to/from the "NoDeSQLog" logical metamodel to the logical schemas for document, key-value, column-family and graph are defined. At this phase, a rule-based algorithm is defined to provide an automated mapping using model driven architecture, and a (large) language model is fine-tuned with context-specific data for testing of mapping learning. The objective is to experiment with the efficiency of LLMs while maximizing the likelihood of the correctness of the output.

(iii) Physical Modeling Phase: From Logical NoSQL Data Models to Generated NoSQL Physical Schemas

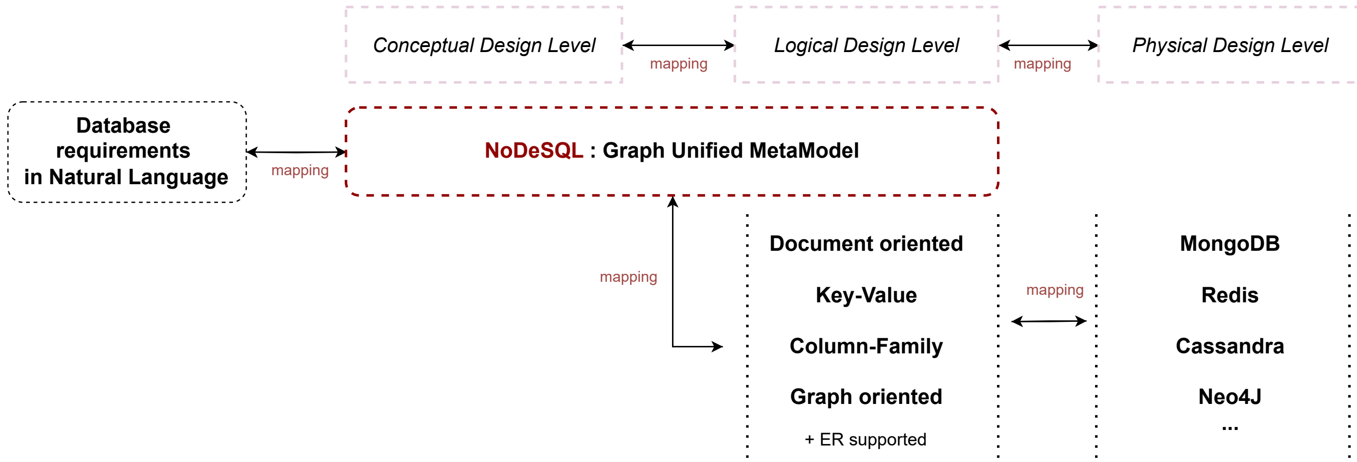

Fig. 1. RQ1: Process integration of the generation of data models from user requirements specified in natural language

In this phase, the corresponding logical Document, Key-Value, Column Family and Graph are used to generate physical schemas in a target database (e.g., MongoDB, Neo4j). Two approaches are tested, one using Neo-EMF, a multi-persistence layer for EMF supporting NoSQL, and the second using fine-tuned (large) language model to generate the physical schema specification as well as JSON schemas.

B. RQ2: AI for NoSQL Querying

This component has two phases: *(i)* the 'translation' from a question/prompt expressed in natural language (NL) to a generic meta-query representation *"Lingua Franca"* (LFQ), and the subsequent *(ii)* mapping from this generic representation to a query expressed in a target query language for a target NoSQL database.

(i) Generation of Lingua Franca Generic Query from Natural Language Prompt

In order to transform a question or prompt in natural language, we build a NER which tags the verbs that represent one of the operations CRUD(Create, Read, Update, Delete), the words that represent an "entity" and the parameters mentioned. A BERT-based operation extraction model is also used. For example: Create (Insert, Add, Create, Make, Build, Generate, Construct, Set up, Establish, Form, etc.), Read (Retrieve, Get, Fetch, Obtain, Show, Display, View, Read, Access, Explore), Update(Modify, Change, Edit, Alter, Update, Revise, Amend, Adapt, Refine, Revamp), Delete(Remove, Erase, Delete, Clear, Eliminate, Exclude, Purge, Discard, Wipe, Obliterate). The generic representation of the query is graph-based. Example: Operation –¿[Node –¿ Edge –¿ Node–¿[Parameter]]

(ii) Generation of NoSQL Query in Target Query Language

After the generation of the LFQ, a translation logic is defined to generate the appropriate syntax in a target database (e.g., SET, GET, DELETE). We compare multiple approaches in this phase, a rule-based approach, a BERT-based model and a fine-tuned large language model. We implement a correctness check syntax correctness and consistency with the schema. The implementation supports multiple query language, and can be schema-aware or schema-independent. If we provide the system with the query, we can have a feedback process for corrections. It can also inform the optimized execution of the query.

C. RQ3: AI for NoSQL Quality Characterization

This component has two parts: (i) Usability study of an Entity-Relationship quality assesment framework for NoSQL , and the subsequent (ii) conceptualization of a NoSQL database-agnostic multi-stakeholder quality characterization framework.

This component comprises a set of experiments harnessing crowdsourcing, expert heuristics and machine learning for the characterization of quality in the context of NoSQL.

(i) Investigation of the perceived usability of an Entity-Relationship quality assessment framework for NoSQL

A qualitative study is conducted with experts to investigate the perceived usability of the Moody-Shanks quality framework, defined for Entity-relationship models, in the context of NoSQL. The seven quality criteria (flexibility, correctness, simplicity, implementability, understandability, completeness, integration) are illustrated using a collected and classified set of Stack Overflow questions and answers under the tag #nosql. A set of experts uses a schema visualization tool and evaluates the schema-set in respect to the quality criteria. A decision tree is constructed based on the results, and trade-off diagrams are generated to illustrate the dependencies of the individual quality criteria and their impact on overall schema quality. Experiment shows the need for a more expressive and complete quality framework.

(ii) NoSQuaLitas: A database-agnostic multi-stakeholer Framework for the characterization of perceived quality in the context of NoSQL data models

An integrative systematic review method is use to identify quality criteria in the field of databases, data quality, data modeling and software engineering. An experiment is performed to understand the most impactful features on overall

Fig. 2. Process integrating the generationg a NoSQL target database query from natural language

model quality. Quality criteria are mapped according to stakeholders' level of expertise required, area of expertise, area of involvement (level of design), and level of involvement. Experts, intermediates and novices are used in an assessment experiment to reflect diverse stakeholder perspectives. Participants use schema visualization and score publicly collected schemas with respect to the quality criteria. Multiple machine learning models are used to infer feature importance and other explainability aspects. A bayesian belief network is used to infer a directed acyclic graph reflecting the feature dependencies. Based on the results, an optimized framework (NoSQuaLitas) is proposed for the characterization of NoSQL data model quality, following the three levels of representation and several stakeholder perspectives.

This component could also contribute in the conceptualization of a quality evaluation assistant integrated into the data model design process. Using a Knowledge Base for Stack Overflow questions and answers, and by comparing a BERT-model with a LLM. This quality assistant could help in the design process, and would be integrated into the mappings between conceptual, logical and physical models, where stakeholder perspective is translated into representation levels, thus allowing for an end-to-end characterization of NoSQL data models.

V. Expected Contributions and Evaluation

This work is expected to produce several contributions, including publications for the quality characterization experiments and proposed framework, the natural language to NoDeSQL unified metamodel and mappings pipeline, the lingua franca query generation pipeline, and a quality assistant for NoSQL database design proof-of-concept and prototype.

To evaluate the overall approach of this work, I plan to perform a case study based on the guidelines of Kitchenham et al [26]. I plan to conduct a study with industrial input to assess the challenges directly in practice. Real-world running examples will be developed to serve as a common ground for tests and discussions. To further evaluate the proposed method, I plan to implement it in a proof-of-concept implementation. The implementation process shall thereby be driven by prototyping to assure and continuously evaluate with the partners that the identified challenges are ad-dressed. Correctness of the query outputs will be evaluated by execution testing. An expert case study will also be performed by the end of the work to assess the usability of the overall system, and its practical adoption in practice. Additionally, language metrics will be evaluated and compared with other query languages.

VI. Current Status and Conclusion

In summary, this work proposes a process comprising the three phases of modeling and their inter-mappings. First, a unified conceptual and logical metamodel is proposed: NoDeSQL and NoDeSQLog. This metamodel is graph-based, flexible, intuitive and is formalized as a property graph with hypergraph extensions. The use of user requirements in natural language offers all stakeholders increased involvement, including business-side and non-specialists. The flexibility of this metamodel offers a range of advantages including expressiveness. Additionally, in most works aiming to propose a uniform design methodology for NoSQL, an aggregate-oriented approach is taken and consequently prioritizes aggregate-oriented databases. Graphs are generally difficult to include in uniform modeling given their inherent difference. Our approach stipulates that instead of unifying data models and then trying to map graphs into a less flexible form of schema, it would be more beneficial to map all schemas to a graph given their understandability and visual aspect. Second, a query generation process is proposed to translate natural language prompts into queries in a target database. And, last, a quality characterization framework is proposed, with experiments based on crowdsourcing, machine learning and stack overflow data.

At present time, I am at different stages of my PhD research depending on the research questions. The experiments for RQ3 have been concluded and data has been analyzed. Implementation of the process outlined in RQ2 is ongoing and analysis of the results of comparisons between different models used will start in the coming months. As for RQ1, the generic model has been formalized, but the mappings are still being developed. Multiple models are being implements for comparison, but results have not been achieved yet.

Acknowledgment

C.A. would like to extend thanks to Pr. Karim Baïna and Pr. Mounir Ghogho for the supervision of this PhD thesis, for their continuing support, and for going above and beyond. C.A. would also like to thank Pr. Abdelatif Benattallah and Pr. Belhajjame Khalid for their collaboration.

References

[1] Chaimae Asaad, Karim Baïna, and Mounir Ghogho. Nosql databases: yearning for disambiguation. *arXiv preprint arXiv:2003.04074*, 2020.

[2] Michael Stonebraker. Sql databases v. nosql databases. *Communications of the ACM*, 53(4):10–11, 2010.

[3] Michael Stonebraker. Stonebraker on nosql and enterprises. *Communications of the ACM*, 54(8):10–11, 2011.

[4] Rick Cattell. Scalable sql and nosql data stores. *Acm Sigmod Record*, 39(4):12–27, 2011.

[5] Bogdan George Tudorica and Cristian Bucur. A comparison between several nosql databases with comments and notes. In *2011 RoEduNet international conference 10th edition: Networking in education and research*, pages 1–5. IEEE, 2011.

[6] Jing Han, Ee Haihong, Guan Le, and Jian Du. Survey on nosql database. In *2011 6th international conference on pervasive computing and applications*, pages 363–366. IEEE, 2011.

[7] Chaimae Asaad and Karim Baïna. Nosql databases–seek for a design methodology. In *Model and Data Engineering: 8th International Conference, MEDI 2018, Marrakesh, Morocco, October 24–26, 2018, Proceedings 8*, pages 25–40. Springer, 2018.

[8] Harley Vera-Olivera, Ruizhe Guo, Ruben Cruz Huacarpuma, Ana Paula Bernardi Da Silva, Ari Melo Mariano, and Maristela Holanda. Data modeling and nosql databases-a systematic mapping review. *ACM Computing Surveys (CSUR)*, 54(6):1–26, 2021.

[9] Kwangchul Shin, Chulhyun Hwang, and Hoekyung Jung. Nosql database design using uml conceptual data model based on peter chen's framework. *International Journal of Applied Engineering Research*, 12(5):632–636, 2017.

[10] Sabrina Sicari, Alessandra Rizzardi, and Alberto Coen-Porisini. Security&privacy issues and challenges in nosql databases. *Computer Networks*, 206:108828, 2022.

[11] Shreya Banerjee and Anirban Sarkar. Ontology driven meta-modeling for nosql databases: a conceptual perspective. *International Journal of Software Engineering and Its Applications*, 10(12):41–64, 2016.

[12] Fatma Abdelhedi, Amal Ait Brahim, Faten Atigui, and Gilles Zurfluh. Logical unified modeling for nosql databases. In *19th International Conference on Enterprise Information Systems (ICEIS 2017)*, pages pp–249, 2017.

[13] Harley Vera, W Boaventura, M Holanda, V Guimaraes, and F Hondo. Data modeling for nosql document-oriented databases. In *CEUR Workshop Proceedings*, volume 1478, pages 129–135, 2015.

[14] Paolo Atzeni, Francesca Bugiotti, and Luca Rossi. Uniform access to non-relational database systems: The sos platform. In *Advanced Information Systems Engineering: 24th International Conference, CAiSE 2012, Gdansk, Poland, June 25-29, 2012. Proceedings 24*, pages 160–174. Springer, 2012.

[15] Paolo Atzeni, Francesca Bugiotti, and Luca Rossi. Uniform access to nosql systems. *Information Systems*, 43:117–133, 2014.

[16] Carlos J Fernández Candel, Diego Sevilla Ruiz, and Jesús J García-Molina. A unified metamodel for nosql and relational databases. *Information Systems*, 104:101898, 2022.

[17] Carlos Javier Fernández Candel, Jesús Joaquín García Molina, and Diego Sevilla Ruiz. Skiql: A unified schema query language. *arXiv preprint arXiv:2204.06670*, 2022.

[18] Kazi Mojammel Hossen, Mohammed Nasir Uddin, Minhazul Arefin, and Md Ashraf Uddin. Bert model-based natural language to nosql query conversion using deep learning approach. *International Journal of Advanced Computer Science and Applications*, 14(2), 2023.

[19] Sebastian Blank, Florian Wilhelm, Hans-Peter Zorn, and Achim Rettinger. Querying nosql with deep learning to answer natural language questions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9416–9421, 2019.

[20] Jichuan Zeng, Xi Victoria Lin, Caiming Xiong, Richard Socher, Michael R Lyu, Irwin King, and Steven CH Hoi. Photon: A robust cross-domain text-to-sql system. *arXiv preprint arXiv:2007.15280*, 2020.

[21] Karthik Radhakrishnan, Arvind Srikantan, and Xi Victoria Lin. Colloql: Robust cross-domain text-to-sql over search queries. *arXiv preprint arXiv:2010.09927*, 2020.

[22] John Klein, Ian Gorton, Neil Ernst, Patrick Donohoe, Kim Pham, Chrisjan Matser, and CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST. Quality attribute-guided evaluation of nosql databases: An experience report. *Pittsburgh, PA: Software Engineering Institute Carnegie Mellon University*, 2014.

[23] John Klein, Ian Gorton, Neil Ernst, Patrick Donohoe, Kim Pham, and Chrisjan Matser. Performance evaluation of nosql databases: a case study. In *Proceedings of the 1st Workshop on Performance Analysis of Big Data Systems*, pages 5–10, 2015.

[24] Paola Gómez, Claudia Roncancio, and Rubby Casallas. Towards quality analysis for document oriented bases. In *Conceptual Modeling: 37th International Conference, ER 2018, Xi'an, China, October 22–25, 2018, Proceedings 37*, pages 200–216. Springer, 2018.

[25] Stefanie Scherzinger and Sebastian Sidortschuck. An empirical study on the design and evolution of nosql database schemas. In *Conceptual Modeling: 39th International Conference, ER 2020, Vienna, Austria, November 3–6, 2020, Proceedings 39*, pages 441–455. Springer, 2020.

[26] Barbara Kitchenham, Lesley Pickard, and Shari Lawrence Pfleeger. Case studies for method and tool evaluation. *IEEE software*, 12(4):52–62, 1995.