Investigating the Perceived Usability of Entity-Relationship Quality Frameworks for NoSQL Databases

Chaimae Asaad[1,2(✉)], Karim Baïna[1], and Mounir Ghogho[2,3]

[1] Alqualsadi, Rabat IT Center, ENSIAS, Mohammed V University in Rabat, Rabat, Morocco
chaimae.asaad@uir.ac.ma

[2] TicLab, Faculty of Engineering and Architecture, International University of Rabat, Rabat, Morocco

[3] University of Leeds, Leeds, UK

Abstract. Quality assessment of data models can be a challenging task due to its subjective nature. For the schemaless, heterogeneous and diverse group of databases falling under the NoSQL umbrella, quality is generally operation and performance oriented, and no quality assessment framework exists. As a first step in shaping our understanding of NoSQL database model quality, this paper investigates the perceived usability of quality evaluation frameworks adopted from Entity Relationship (ER) modeling to the context of NoSQL databases. A first evaluation is performed on the three most widely used ER quality frameworks, where they are assessed for their usefulness, ease of use and suitability in the context of NoSQL databases. Based on the results of this assessment, a second evaluation is performed on the best scoring framework. This evaluation is comprised of a real use case adoption of the framework to assess the quality of NoSQL database models. This paper merges targeted crowdsourcing, Stack overflow data mining and white-box classification to gain insights into the concept of NoSQL database model quality, its characterizing features and the trade-offs it involves. This work illustrates the first investigation of ER-defined quality framework to NoSQL on a sample of diverse NoSQL schemas and using both industrial and academic participants. A decision tree is utilized to describe the heuristics of data model assessment, and an analysis is performed to identify inter-annotator disagreement, quality criterion importance, and quality trade-offs. In the absence of works approaching NoSQL data model quality assessment, this paper aims to lay groundwork and present preliminary insights on quality characterization in the context of NoSQL, as well as highlight current gaps, limitations and potential improvements.

Keywords: NoSQL Databases · Data Model Quality Assessment · Crowdsourcing · Decision Trees

M. Mosbah et al. (Eds.): MEDI 2023, LNCS 14396, pp. 214–227, 2024.
https://doi.org/10.1007/978-3-031-49333-1_16

1 Introduction

Quality assessment represents a salient phase for information systems [1–4], software design and engineering [5–7], business Process models [8–10], conceptual [6,10–13] and logical design [14,15]. In the process of database design, frameworks have been proposed to assess quality of conceptual [11,12], logical [14,16] and physical schemas [17]. NoSQL database quality evaluation mainly focuses on aggregate-oriented NoSQL databases (Key-Value, Document, Column Family), and targets physical-level attributes such as transaction performance, partition-tolerance and data model mapping [18,19], availability and security [17], and consistency, performance, scalability [20]. In addition to availability and security, popularity, maturity, query possibilities, concurrency control and conflict resolution were considered as quality attributes in a proposed framework aiming to assist IT departments align perceived risks of NoSQL database adoption [17]. Works approaching NoSQL data model quality from a logical design perspective are scarce, and often focus on one category of NoSQL database. For instance, a set of metrics including types, collections, nesting depth, width of documents, referencing rate and redundancy were proposed to characterize aspects of the complexity of document-oriented schemas with the aim of facilitating schema analysis and comparison [21]. The scarcity of literature relating to the task of evaluating quality for NoSQL database models is due to the flexible characteristics of NoSQL databases [22]. An a priori defined schema is often not required for ingesting data into a NoSQL database. Additionally, NoSQL databases are highly heterogeneous and differ in type, features and underlying data model [22]. The lack of standardization further complicates quality assessment [23]. In the absence of a framework addressing database-agnostic quality for NoSQL databases, the gap in literature remains. In an effort to approach the quality assessment of data models for NoSQL databases, this paper investigates the potential applicability of Entity-Relationship (ER) quality frameworks to evaluate NoSQL database models. Several works have proposed frameworks and quality criteria aiming to evaluate Entity-Relationship models. For instance, Genero et al. [24] proposed and validated different ERD structural complexity measures such as number of entities, number of derived attributes, number of composite attributes, etc. In this work, we evaluate three ER quality frameworks identified in a study presented by Krogstie et al. [25] as the most cited frameworks: the Moody and Shanks framework [26], and the Batini and Scannapieco framework [16]. We add to the evaluation the Kesh Someswar framework [14] to allow for further comparison. In the remainder of this paper, we refer to these three frameworks as the MS (Moody Shanks), BS (Batini Scannapieco) and KS (Kesh Someswar) frameworks. In this paper, we first perform an experiment aiming to evaluate the perceived usability of the three aforementioned ER quality framework for NoSQL database models[1] using experts from both industrial and

[1] In this work, database model, data model and schema are used interchangeably given the context of physical design in NoSQL databases. These concepts are, however, not equivalent in other contexts of database design.

academic communities. We defined perceived usability based on three variables: perceived ease of use (PEU), perceived usefulness (PU) and perceived suitability (PS). Based on the results of the first experiment, we identified the most well scored framework and performed a real use-case adoption of the framework and an analysis of its quality criteria using data mining on Stack Overflow and decision trees. Obtaining ratings from participants has been previously employed in the literature. For instance, quality sub-characteristics of Entity-Relationship diagrams were rated using a scale of 7 linguistic labels by a group of subjects [24]. Stack Overflow questions have previously been mined in the literature to investigate the main challenges and issues faced by developers of NoSQL databases [27]. In machine learning, crowdsourcing is a popular method to acquire ground truth [28]. However, in many of its applications areas, extreme difficulties can be met in obtaining such ground truth, in most part due to high costs and task subjectivity [29]. Questionnaires were previously used in the literature in similar contexts. For instance, in [24], a questionnaire was used for the evaluation of participants' level of understanding of the entity relationship diagrams to be rated. The contributions of this paper are manifold and include a first investigation of potential adoption and adaptability of ER quality frameworks to the NoSQL context, an analysis of an ER framework's applicability in a real case scenario, the use of a hybrid method comprised of data mining, targeted crowdsourcing and decision trees for the identification of quality criterion importance, quality trade-offs and quality characterization heuristics. A presentation of threats to validity and potential improvements is included to guide future scaled up experiments. The remainder of this paper is organized as follows: Sect. 2 presents our the first experiment proposed, along with its results and analyses performed. In Sect. 3, the testing of the best scoring quality framework is conducted, and results are highlighted. In Sect. 4, discussions into potential applicability, improvements as well as threats to validity are detailed. Conclusions and future work are presented in Sect. 5.

2 Perceived Usability of the MS, BS and KS Frameworks

In this section, we propose and perform an experiment using targeted crowdsourcing to gauge the 'perceived usability' of three Entity-Relationship (ER) frameworks, Moody Shanks (MS) [26], Batini Scannapieco (BS) [16], and Kesh Someswar (KS) [14], in a NoSQL database context.

2.1 Background

The Moody-Shanks data model quality framework [26,30] was conceptualized for the quality evaluation of Entity Relationship (ER) data models. This framework includes seven quality criteria: Correctness, Completeness, Simplicity, Flexibility, Integration, Understandability, and Implementability. In the context of Entity-Relationship data models, the Moody-Shanks framework defined *correctness* as the "conformity to the modeling rules and techniques", *completeness* as "the representation of all information included in user requirements", *simplicity* as the

"minimality of representative structures and lack of unnecessary components", *flexibility* as the "model's adaptability to future changes", *understandability* as "the ease of understanding of the model", *integration* as the "consistency of the model within the context it is defined", and *implementability* as "the ease, cost, and time consumption facets of model implementation" [26,30]. The Batini Scannapieco framework [16] approached the issue of improving the quality of a database schema, in addition to schema documentation, implementability and maintenance, and provided a set of quality criteria: Completeness, correctness, minimality, expressiveness, readability, self explanation, extensibility and normality. In this framework, a schema is complete when "it represents all relevant features of the application domain", correct when "it properly uses the concepts of the ER model, syntactically and semantically", minimal when "every aspect of the requirements appears only once in the schema and no concept can be deleted without loss of information", expressive when "it represents requirements in a natural way and can be easily understood", readable when "it respects certain aesthetic criteria that make the diagram graceful" (e.g., drawn in a grid, minimal number of crossings, etc.), self explanatory when "a large number of properties can be represented using the conceptual model itself, without other formalisms", extensible when "it is easily adapted to changing requirements", and normalized based on the theory of normalization associated with the relational model. The Kesh Someswar framework [14] differentiated between ontological and behavioral attributes in quality assessment of an E-R model. In this context, ontological quality is defined based on two facets, quality of structure and quality of content, where the former includes as criteria suitability, soundness, consistency and conciseness, and the latter includes completeness, cohesiveness and validity. Behavioral quality, on the other hand, includes user usability, designer usability, maintainability, accuracy and performance. In the context of this framework, suitability of structure refers to "the fact that form should follow structure", soundness represents "adherence to technical design principles", consistency and conciseness represent respectively the lack of contradictions in the model" and "redundant relationships". Completeness refers to the inclusion of "all attributes of the entities", cohesiveness to the "closeness of attributes", validity to the "correct representation of descriptions and properties of the attributes". Usability is defined for users and designers respectively as the extent to which "users will feel confident from their diagram that requirements were taken care" and for "designers to proceed to the next stage". Maintainability represents the "ease with which the model can be modified, corrected and extended", while accuracy represents "the reliability of the model" and performance reflects its "efficiency".

2.2 Experiment

In order to gauge the perceived usability of the three aforementioned quality frameworks for the NoSQL context, we propose the following experiment.

We first define perceived usability based on three variables: perceived ease of use (PEU), perceived usefulness (PU) and perceived suitability (PS). We adopt the formalization of perceived usefulness and perceived ease of use from the

application of the Technology Acceptance Model [31,32] which describes how the actual system's usage depends on the attitude of users and defines measurement scales based on which these variables can be evaluated, and which is widely applied for different information systems [33]. Additionally, we define the perceived suitability as an extra measure to take into account the framework's potential suitability for the NoSQL context. Together, these three variables reflect perceived usability. For PU, we define 6 characterizing items, based on application of the Technology Acceptance Model [31,32]: work more quickly, job performance, increase productivity, effectiveness, makes job easier, useful. For PEU, we define 6 characterizing items, based on application of the Technology Acceptance Model [31,32]: easy to learn, controllable, clear and understandable, flexible, easy to become skillful, easy to use. For PS, we define 3 characterizing items: relevance to NoSQL, representativeness of NoSQL, willingness to apply in real case scenarios. The next step includes using targeted crowdsourcing, we use volunteering annotators to score the three frameworks for each of the characterizing items of PEU, PU and PS. Unlike its conventional counterpart, targeted crowdsourcing requires a specific type of workers for tasks that are either subjective or knowledge intensive [34]. In software engineering, expert opinion is the most frequently used validation method [28]. In the case of quality criteria annotation, the task is subjective both by design and by nature, and thus the use of multiple sources of annotations presents a perfect opportunity for a "natural shift from the traditional reliance on a single domain expert [29]". Given the subjectivity and high expertise required in data model quality and quality criteria annotation, multi-annotator targeted crowdsourcing was selected as the most appropriate method. Given the different stakeholder perspectives involved in the design process of any application or data model, quality evaluation is highly dependant on which perspective is taken into consideration and which stakeholder's satisfaction is prioritized. In this perceived usability study, we exclusively focus on the architect's and builder's perspectives. The architect's perspective represents the view of the data modeler or analyst responsible for developing the data model and ensuring its conformance to data modeling practices [26]. The builder's perspective, on the other hand, represents the view of the 'application' developer responsible for the implementation of the data model in a particular technology [26]. These two perspectives constitute what we denote "expert perspective", and represent the view of the expert responsible for the data modeling and implementation of the data model in a particular NoSQL database. For NoSQL databases, the modeling phase is integrated into the "implementation" phase given the schema-less nature of NoSQL [22], and so, the perspectives of the architect and builder are often enmeshed. In order to gain expert PEU, expert PU and expert PS, we use a skills filtering process given that a high level of expertise is required. To diversify the pool of annotators participating in this experiment, we include both the industrial and academic communities. In order to objectively capture level of expertise, we use classifications provided by both the 'Multilingual Classification of

European Skills, Competences, Qualifications and Occupations (ESCO)[2]' [35] and SkillsDB[3]. Data obtained from ESCO and SkillsDB respectively allows for a mapping between academic and industrial fulfillment of required NoSQL expertise. We evaluated the experts involved in the experiment based on these skill classifications. Experts were required to have background and experience corresponding to at least 80% of these skills to qualify for participation. As a result, we include 37 volunteer annotators, 17 of which are from the academic community and 17 from the industrial community. A 5 Likert scale is used in this experiment to score each characterizing item of perceived ease of use (PEU), perceived usefulness (PU) and perceived suitability (PS), reflecting the following scale: extremely unlikely, slightly unlikely, neither likely or unlikely, slightly likely, extremely likely. To annotate, the participants are instructed to formulate the question as "Would using the quality framework [X_i] allow for [characterizing_item]?", where X_i refers respectively to the MS, BS and KS framework. To avoid bias propagation and experiment contamination, the expert annotators were instructed not to discuss the experiment or the results of the annotation with one another.

2.3 Results

Upon the performance of the annotation experiment, we found that the average annotations for the MS framework are consistently higher in most characterizing features comprising the definition of perceived ease of use, perceived usefulness and perceived usability. In order to assess the statistical significance of these results, we perform the Mann-Whitney U test comparing the distributions of the annotations for Group "MS" and the combined groups "BS" and "KS" over the groups PEU, PU, and PS. In this case, we find that the p-values for all three feature groups (PEU, PU, and PS) are below 0.05, indicating that there is a significant difference between the distributions of the annotations for the frameworks MS and the combination of BS and KS. This result is consistent for the two sample T test comparing the means of the annotations of MS and the combined BS and KS groups over feature groups PEU, PU and PS. The same result is found when the U test is performed on the characterizing items. For most of the PEU features and some of the U and S features, the p-values are below 0.05, indicating that there is a significant difference between the distributions of the annotations, however, for some features (PU1, PU2, PU3, PU6, PS1, PS2), the p-values are above 0.05, suggesting that there is no significant difference between the distributions of the annotations for these features.

3 MS as a Quality Framework for NoSQL Databases

Based on the results of the previous section, we identify that the Moody Shanks MS framework was annotated at a higher score across most characterizing items

[2] https://esco.ec.europa.eu/.
[3] https://www.skillsdb.net/.

with a statistical significance. This means that the expert annotators found the MS framework to be easy to use, useful and suitable for the NoSQL context. In order to further investigate its applicability in a real use case, we conduct the following experiment.

3.1 Experiment

In this experiment, our aim is to put the MS framework to the test and explore its potential applicability by real practitioners in a quality assessment scenario. To that end, publicly available NoSQL schemas were collected from different sources such as database websites, modeling websites, etc. The NoSQL schema dataset is constituted of 35 schemas of different NoSQL databases. The database is comprised of 51.4% document oriented database models, 28.6% graph oriented database models, 14.3% key value database models and 5.7% column-family database models. These models are implemented in different databases: Cassandra, Couchbase, DynamoDB, MongoDB and Neo4j.

These models will be scored with respect to the 7 quality criteria outlined in the MS framework: Correctness, Completeness, Simplicity, Flexibility, Integration, Understandability and Implementability. A Likert scale ranging from 1 to 10 is used to score each quality criterion based on the annotator's level of agreement with the schema's fulfillment of the given criterion. We use 30 of the same annotators to score these measures, while the remaining 7 score the schemas for a measure of *overall quality*, defined to the participants as "as-is use", i.e., "would you use this data model as it is without making any changes, to operate and apply queries, for this particular universe of discourse". Two iterations of this experiment are conducted. The first where the annotators are given the schemas as JSON files and visualizations using Hackolade[4], along with documentation. And the second iteration where concise definitions of the quality criteria along with application examples are provided to the participants. In order to concisely define the quality criteria of the Moody-Shanks MS quality framework, we conduct a data collection process on two fronts. First, we collect modeling guidelines and good (or bad) practices from various NoSQL database providers' websites and technical reports. These guidelines are subsequently organized in documents and are presented to the participants in the form of 'cheat sheets' or 'quick recaps' for data modeling in multiple NoSQL database categories and databases (e.g., MongoDB, Cassandra). The second data collection consists of web scraping Stack Overflow[5] for Questions and Answers (Q&A) related to the NoSQL tag/topic. The collection of good/bad practices, coupled with the collected Stack Overflow Q&A, are used to illustrate the mapping from the Moody-Shanks quality criteria to real use case application examples. The collected Stack Overflow Q&A data was collected by an initial crawling of the #nosql tag, resulting in a dataset of 1020 rows comprised of the following cells: question, answer, author of answer,

[4] https://hackolade.com/.

[5] Stack Overflow is a public platform that provides a community-based space to find and contribute answers to technical challenges. Link: https://StackOverflow.com/.

Table 1. Quality Criteria Definitions and Examples Used in Second Iteration of the Experiment

Quality Criterion	Adapted Redefinition	Application Examples
Correctness	Relates to the appropriate use of structures within data model	- *Using embedding to model an object that will be accessed on its own (MongoDB)* - *Uniqueness of column key and row key to avoid accidental overwriting (Cassandra)*
Completeness	Inclusion of minimum requirements to fulfill functionality	- *Absence of relationships in graph database (Neo4j)* - *Unrepresented access patterns and keys (Cassandra)*
Simplicity	Conciseness and minimality of data model	- *Excessive number of duplicated properties that represent complex data instead of one shared node representing the property (Neo4j)* - *Structural redundancy in embedding documents leading to issues in coherency (MongoDB)*
Flexibility	Existence of structural qualities in the data model enabling (or hindering) ease of evolution	Some design choices potentially hindering flexibility: - *An embedding of sub-documents implying a pre-join (MongoDB).* - *Nested columns or supercolumns and their nesting depth (Cassandra).* - *Storing data in two different buckets or in a single one affects access since sub-objects are not supported (Riak).* - *Ease in schema evolution in terms of node additions, deletions or merging (Neo4j).*
Integration	Relates to the absence of contradictions within data model structures	- *An unconstrained relationship between two nodes contradicting type (Neo4j).* - *Representing a document embedding explicitly contradicting another (MongoDB).*
Understandability	Clarity of the data model for relevant stakeholders	- *Trade-off between modeling complex data and impact on explainability of model*
Implementability	Estimated time constraints and effectiveness of data model in realizing access patterns	- *Modeling data to reduce depth of downstream traversal access path (Neo4j)* - *Modeling downstream data as a relationship instead of node label or property (Neo4j)*

badge(s) of author of answer, votes of answer. This data contained varying percentages of mentions to different NoSQL databases, namely, 27.1% of questions pertaining to mongoDB, 10.29 % questions related to cassandra, around 10% pertaining to dynamoDB (5%) and Redis (5.3%). This dataset was then further extended using the Neo4j Stack Overflow dump database[6] comprised of 10.000.488 programming questions, 16.548.187 solutions and 138.390.250 comments and edits. Because of the diverse perspectives of stakeholders, quality criteria often have varying descriptions and definitions. Additionally, the chosen design level impacts how quality criteria are defined. If the evaluation is done on a requirements level, then the quality criteria definitions will relate to the requirement. Similarly, at the data level, quality criteria will reflect aspects of data quality. In this paper, the scope of the quality assessment is strictly defined

[6] https://archive.org/download/stackexchange.

with respect to the design and implementation level (which we consider as one, given the nature of NoSQL). The requirements analysis and deployment levels, which incorporate requirement quality, meta-model quality, modeling quality and data quality [3] are beyond the scope of this paper. Additionally, all stakeholder perspectives besides the expert perspective (combining the architect's and builder's perspectives) are beyond the scope of this paper. And thus, these are the parameters of evaluation of the adapted quality criteria. The result of this process is illustrated in the table below where each MS quality criterion is concisely presented and highlighted by application examples, thus providing a learning-by-example process for the participants in their annotation task. Statistical analysis is then conducted to compare the results of the two iterations, and a decision tree is used to infer feature importance and quality trade-offs between the criteria.

3.2 Results

To identify the features that are most agreed upon by the annotators, we generated a heatmap and identified the cells with the lowest standard deviation, indicating that the annotations for those features have less variation among the annotators. When taking the example of a schema corresponding to "Buzzfeed", the features "Integration" and "Implementability" have the lowest disagreement among annotators. For the schema "DynamoDB examples", the features "Integration" and "Completeness" have the lowest disagreement among annotators. For the schema "Kansas City Fountains" the features "Understandability" and "Simplicity" have the lowest disagreement among annotators. This pattern continues for the other schemas as well. It's important to note that the standard deviation is a measure of dispersion, so a lower value indicates that the annotations are closer to the mean, and hence, there is more agreement among the annotators. To get an overall view of the features most agreed upon across the entire dataset, we can look at the average standard deviation for each feature. A lower average standard deviation indicates that the annotations for that feature have less variation among the annotators, and thus, there is more agreement. Results show the features "Completeness" and "Flexibility" have the lowest average standard deviation, indicating that these features have the most agreement among annotators across the entire dataset. On the other hand, the feature "Correctness" has the highest average standard deviation, indicating that it has the least agreement among annotators.

Disagreement Analysis. After introducing the concise definitions of the MS quality criteria and the application examples mapped using Stack overflow data (Table 1), the features "Flexibility" and "Implementability" have the lowest average standard deviation in the perturbed data, indicating that these features have the most agreement among annotators. On the other hand, the feature "Simplicity" has now the highest average standard deviation, indicating that it has the least agreement among annotators. The second iteration's average standard deviations are now for flexibility, implementability, integration, completeness,

correctness, understandability and simplicity, 0.917, 1.407, 1.418, 1.427, 1.441, 1.454 and 1.457, respectively. In analyzing the box plot, we found that the variability in annotations has significantly decreased, which can further highlight the benefits of a concise definition of quality criteria and the high impact of having real use case application examples to guide the quality evaluation process.

Decisions Trees for White-box modeling of Expert Opinion. Traditionally, supervised learning employs a domain expert fulfilling the 'teacher' role and thus providing necessary supervision [29]. The most common case being one where expert annotations serve as data point labels in classification problems [29]. In this experiment, labels were obtained through targeted crowdsourcing and subsequently fed to a decision tree model. The objective behind the use of decision trees is not the classification task, but rather the white box (WB) modeling of the heuristics of expert quality assessment. The visual, interpretable, explainable and transparent characteristics of WB models motivated the use of decision trees in this experiment. In the case of NoSQL data model quality assessment experiment, a decision tree is fed the annotations of all 30 participants as data points, while a majority-vote of the "overall quality" annotations of the 7 participants is used to generate labels. The seven quality criteria included in the Moody-Shanks framework and used as features for the decision tree have varying levels of importance and characterize quality with a 75% accuracy. Integration, simplicity, completeness and correctness were found to be the most important features, with importance percentages of 17.39%, 16.26%, 16.16% and 16.08% respectively, while Understandability, Implementability and Flexibility have importance percentages of 12.23%, 10.99% and 10.89% respectively. Using the results of the decision tree, different scenarios can be constructed to illustrate how the degree of fulfillment of these quality criteria affects the overall quality of the data model. In one trade-off example, Implementability seems to outweigh completeness, i.e., a data model can still be of good overall quality even when completeness is not fulfilled, as long as implementability is. In contrast, a data model would be of bad overall quality, if it is neither flexible nor understandable, even if completeness is fulfilled. These trade-offs speak to the interactions between various combinations of quality criteria and overall quality of the data model, and give some insight on the process of quality characterization and assessment for NoSQL databases.

4 Discussion

In this paper, an experiment was proposed for the investigation of the adaptation of three Entity-Relationship quality frameworks for NoSQL database model quality assessment. Preliminary results describing the experiment were presented. A learning-by-example process enabled by Stack Overflow question/answer collection and mapping with good practices and guidelines was tested for its ability and was found to decrease inter-annotator disagreement, even in a subjective task such as quality assessment. Expert annotations were conducted on publicly

available collected NoSQL schemas and a decision tree was leveraged for feature importance and trade-offs. In highly subjective annotation tasks, analysis of inter-annotator agreements and disagreements might make it possible to build classifiers that embody the inter-subjective overlap between the mental conceptions of the annotators [36]. Experiment replication is necessary in order to draw any further conclusions on this aspect of annotation. These preliminary results enable some elucidation of the heuristics used by experts in assessing quality in the context of NoSQL database models, and allow us to shed light on potential improvements to the adaptation of the Moody-Shanks framework in the context of NoSQL databases. Although a concise definition and example illustration of Moody-Shanks quality criteria was conducted in this work, a quantification of these criteria as metrics would add objectivity and allow for an empirical approach to quality assessment. Such metrics need to be NoSQL-specific. A few works in the literature mentioned NoSQL-specific characteristics such as schema size and denormalized schema state [37], embedding and nesting 'levels' [38], normalization and embedding [39] and aggregation [23]. These characteristics can potentially be metrics illustrating quality criteria such as flexibility and correctness. Another perspective on the improvement of the framework delineates the potential completion of the quality criteria of Moody-Shanks framework by additional criteria from other quality frameworks used in conceptual modeling and software design upon adapting them to the NoSQL context. Additionally, linking the quality criteria to universe of discourse requirements, query-first design and data quality aspects may potentially transform the quality framework and allow it to be functional in real use-cases. The findings of this study are to be considered in light of a number of limitations. The work conducted in this paper represents a first investigation of a perceived usability study and is an ongoing effort, and therefore further experiment replication, statistical analysis and hypothesis testing is paramount for any potential generalizability of findings. The subjectivity of the target variables as well as potential annotation bias represent substantial limitations, and although quality evaluation can often be subject to bias, we contend that this work is an effort to understand such subjectivity and not to eliminate it. Various threats to both internal and external validity can be highlighted. Aspects related to internal validity include (i) difference in level of expertise amongst subjects. In this experiment, all participants were evaluated with the condition of fulfilling a minimum of skills outlined by the skill classification knowledge bases used (ESCO and SkillDB) in order to ensure equivalence in skill. The challenge of (ii) knowledge of the universe of discourse was addressed by providing documentation to all participants and including details and additional context on each schema. Threats to external validity include materials used, dataset size and number of participants. These limitations, however, are relevant to the experiment and not the methodology proposed. In the context of NoSQL database model quality assessment, the gap in literature is significant, and consequently, this work aims to lay ground work and present a first effort to approach quality using a methodology combining learning-by-example, targeted crowdsourcing and white-box modeling.

5 Conclusions and Future Work

Characterizing NoSQL data model quality using Entity Relationship frameworks is a novel idea. In his work, we gauged the concept of perceived usability by defining it as the sum of perceived ease of use, perceived usefulness and perceived suitability. Results led to a real use case test of the applicability of the Moody-Shanks framework, with and without explicit application examples. These experiments have yielded insights potentially delineating the process of quality evaluation conducted by experts. Targeted crowdsourcing allowed testing of the usability, ease of use and validity of the framework in the context of NoSQL databases. White box model decision trees allowed for the determination of quality criterion importance and construction of feature trade-off diagrams. Using Stack Overflow question/answer data and mapping them as examples to illustrate the quality criteria allowed for a NoSQL-specific formalization of the Moody-Shanks framework and added a layer of relative objectivity and standardization of the annotation process. Threats to validity were explicitly highlighted and ongoing efforts are aiming to enrich criteria definitions with quantifiable metrics thus allowing for less subjectivity in the framework as well as potential automation.

Acknowledgements. Authors would like to extend thanks to all the participants for their time, efforts and contributions to the experiment.

Data Availibility Statement. All data, schemas, documentation, annotations and further illustrations and graphs used in experimentation are available online: https://github.com/ChaiAsaad/NoSQL-Quality-Experiment.

References

1. Dedeke, A.: A conceptual framework for developing quality measures for information systems. In: IQ, pp. 126–128 (2000)
2. Moody, D.L.: The method evaluation model: a theoretical model for validating information systems design methods. In: ECIS 2003 Proceedings (2003)
3. Thi, T.T.P., Helfert, M.: A review of quality frameworks in information systems. arXiv preprint arXiv:1706.03030 (2017)
4. Batini, C., Scannapieco, M.: Data Quality: Concepts, Methodologies and Techniques. Springer, Heidelberg (2006)
5. Lourenço, J.R., Abramova, V., Vieira, M., Cabral, B., Bernardino, J.: NoSQL databases: a software engineering perspective. In: Rocha, A., Correia, A.M., Costanzo, S., Reis, L.P. (eds.) New Contributions in Information Systems and Technologies. AISC, vol. 353, pp. 741–750. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-16486-1_73
6. Moody, D.L., Sindre, G., Brasethvik, T., Solvberg, A.: Evaluating the quality of information models: empirical testing of a conceptual model quality framework. In: Proceedings of the 25th International Conference on Software Engineering, pp. 295–305. IEEE (2003)
7. Blin, M.-J., Tsoukiàs, A.: Multi-criteria methodology contribution to the software quality evaluation. Softw. Qual. J. **9**(2), 113–132 (2001)

8. Sánchez-González, L., García, F., Ruiz, F., Piattini, M.: Toward a quality framework for business process models. Int. J. Cooperative Inf. Syst. **22**(01), 1350003 (2013)
9. Moody, D.L., Sindre, G., Brasethvik, T., Sølvberg, A.: Evaluating the quality of process models: empirical testing of a quality framework. In: Spaccapietra, S., March, S.T., Kambayashi, Y. (eds.) ER 2002. LNCS, vol. 2503, pp. 380–396. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45816-6_36
10. Moody, D.L.: Theoretical and practical issues in evaluating the quality of conceptual models: current state and future directions. Data Knowl. Eng. **55**(3), 243–276 (2005)
11. Eick, C.F.: A methodology for the design and transformation of conceptual schemas. In: VLDB, vol. 91, pp. 25–34 (1991)
12. Cherfi, S.S.-S., Akoka, J., Comyn-Wattiau, I.: Measuring UML conceptual modeling quality, method and implementation. In: Pucheral, P. (ed.) Proceedings of the BDA Conference, Collection INT, France (2002)
13. Shanks, G., et al.: Conceptual data modelling: an empirical study of expert and novice data modellers. Australas. J. Inf. Syst. **4**(2) (1997)
14. Kesh, S.: Evaluating the quality of entity relationship models. Inf. Softw. Technol. **37**(12), 681–689 (1995)
15. Moody, D.L., Shanks, G.G., Darke, P.: Improving the quality of entity relationship models—experience in research and practice. In: Ling, T.-W., Ram, S., Li Lee, M. (eds.) ER 1998. LNCS, vol. 1507, pp. 255–276. Springer, Heidelberg (1998). https://doi.org/10.1007/978-3-540-49524-6_21
16. Batini, C., Ceri, S., Navathe, S.B., et al.: Conceptual Database Design: An Entity-Relationship Approach, vol. 116. Benjamin/Cummings, Redwood City (1992)
17. Mackin, H., Perez, G., Tappert, C.C.: Adopting NoSQL Databases Using a Quality Attribute Framework and Risks Analysis. SCITEPRESS - Science and Technology Publications, Lda. (2016)
18. Klein, J., Gorton, I., Ernst, N., Donohoe, P., Pham, K., Matser, C.: Quality attribute-guided evaluation of NoSQL databases: an experience report. Technical report, Carnegie-Mellon Univ Pittsburgh PA Software Engineering Inst (2014)
19. Klein, J., Gorton, I., Ernst, N., Donohoe, P., Pham, K., Matser, C.: Performance evaluation of NoSQL databases: a case study. In: Proceedings of the 1st Workshop on Performance Analysis of Big Data Systems, pp. 5–10. ACM (2015)
20. Klein, J., Gorton, I.: Design assistant for NoSQL technology selection. In: 2015 1st International Workshop on Future of Software Architecture Design Assistants (FoSADA), pp. 1–6. IEEE (2015)
21. Gómez, P., Roncancio, C., Casallas, R.: Towards quality analysis for document oriented bases. In: Trujillo, J.C., et al. (eds.) ER 2018. LNCS, vol. 11157, pp. 200–216. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00847-5_16
22. Asaad, C., Baïna, K., Ghogho, M.: NoSQL databases: yearning for disambiguation. arXiv e-prints arXiv:2003.04074 (2020)
23. Asaad, C., Baïna, K.: NoSQL databases – seek for a design methodology. In: Abdelwahed, E.H., Bellatreche, L., Golfarelli, M., Méry, D., Ordonez, C. (eds.) MEDI 2018. LNCS, vol. 11163, pp. 25–40. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00856-7_2
24. Genero, M., Piattini, M., Calero, C.: Assurance of conceptual data model quality based on early measures. In: Proceedings Second Asia-Pacific Conference on Quality Software, pp. 97–103. IEEE (2001)
25. Krogstie, J.: Quality of conceptual data models. In: ICISO 2013 (2013)

26. Moody, D.L., Shanks, G.G.: What makes a good data model? Evaluating the quality of entity relationship models. In: Loucopoulos, P. (ed.) ER 1994. LNCS, vol. 881, pp. 94–111. Springer, Heidelberg (1994). https://doi.org/10.1007/3-540-58786-1_75
27. Islam, S., Hasan, K., Shahriyar, R.: Mining developer questions about major NoSQL databases. Int. J. Comput. Appl. **975**, 8887 (2021)
28. Yan, M., Xia, X., Zhang, X., Xu, L., Yang, D.: A systematic mapping study of quality assessment models for software products. In: 2017 International Conference on Software Analysis, Testing and Evolution (SATE), pp. 63–71. IEEE (2017)
29. Yan, Y., et al.: Modeling annotator expertise: learning when everybody knows a bit of something. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, pp. 932–939. JMLR Workshop and Conference Proceedings (2010)
30. Moody, D.L., Shanks, G.G.: What makes a good data model? A framework for evaluating and improving the quality of entity relationship models. Aust. Comput. J. **30**(3), 97–110 (1998)
31. Davis, F.D.: A technology acceptance model for empirically testing new end-user information systems: theory and results. Ph.D. thesis, Massachusetts Institute of Technology (1985)
32. Davis, F.D.: Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS Q. 319–340 (1989)
33. Jalali, A.: Evaluating user acceptance of knowledge-intensive business process modeling languages. Softw. Syst. Model. 1–24 (2023)
34. Yang, J., Drake, T., Damianou, A., Maarek, Y.: Leveraging crowdsourcing data for deep active learning an application: learning intents in Alexa. In: Proceedings of the 2018 World Wide Web Conference, pp. 23–32. International World Wide Web Conferences Steering Committee (2018)
35. De Smedt, J., le Vrang, M., Papantoniou, A.: ESCO: towards a semantic web for the European labor market. In: Ldow@ WWW (2015)
36. Reidsma, D., op den Akker, R.: Exploiting 'subjective' annotations. In: Coling 2008: Proceedings of the workshop on Human Judgements in Computational Linguistics, pp. 8–16 (2008)
37. Scherzinger, S., Sidortschuck, S.: An empirical study on the design and evolution of NoSQL database schemas. arXiv preprint arXiv:2003.00054 (2020)
38. Vera, H., Boaventura, W., Holanda, M., Guimaraes, V., Hondo, F.: Data modeling for NoSQL document-oriented databases. In: CEUR Workshop Proceedings, vol. 1478, pp. 129–135 (2015)
39. Kanade, A., Gopal, A., Kanade, S.: A study of normalization and embedding in MongoDB. In: 2014 IEEE International Advance Computing Conference (IACC), pp. 416–421. IEEE (2014)