Expert Systems

WILEY

Expert Systems

REVIEW

The Challenge of Generating and Evolving Real-Life Like Synthetic Test Data Without Accessing Real-World Raw Data—A Systematic Review

Maj-Annika Tammisto[1] | Faiz Ali Shah[1] | Daniel Rodriguez[2] | Dietmar Pfahl[1]

[1]Institute of Computer Science, University of Tartu, Tartu, Estonia | [2]Department of Computer Science, University of Alcala, Madrid, Spain

Correspondence: Maj-Annika Tammisto (maj-annika.tammisto@ut.ee)

Received: 8 November 2024 | **Revised:** 10 September 2025 | **Accepted:** 24 October 2025

Funding: This work was partly funded by BMK, BMAW, and the State of Upper Austria in the frame of the SCCH competence center INTEGRATE [(FFG grant no. 892418)] part of the FFG COMET Competence Centers for Excellent Technologies Programme, as well as by the European Regional Development Fund, grant PRG1226 of the Estonian Research Council, and the JCLM project SBPLY/24/180225/000143 under the European Regional Development Fund (FEDER).

Keywords: artificial data | data evolution | data synthesis | synthetic data generation | synthetic test data | synthetically generated data

ABSTRACT

Background: High-level system testing of applications that use data from e-Government services as input requires test data that is real-life-like but where the privacy of personal information is guaranteed. Applications with such strong requirement include information exchange between countries, medicine, banking, and so on. This review aims to synthesise the current state-of-the-practice in this domain.

Objectives: The objective of this Systematic Review is to identify existing approaches for creating and evolving synthetic test data without using real-life raw data.

Methods: We followed well-known methodologies for conducting systematic literature reviews, including the ones from Kitchenham and PRISMA as well as guidelines for analysing the limitations of our review and its threats to validity.

Results: A variety of methods and tools exist for creating privacy-preserving test data. Our search found 1013 publications in IEEE Xplore, ACM Digital Library, and SCOPUS. We extracted data from 75 of those publications and identified 37 approaches that answer our research question partly. A common prerequisite for using these methods and tools is direct access to real-life data for data anonymization or synthetic test data generation. Nine existing synthetic test data generation approaches were identified that were closest to answering our research question. Nevertheless, further work would be needed to add the ability to evolve synthetic test data to the existing approaches.

Conclusions: None of the publications covered our requirements completely, only partially. Synthetic test data evolution is a field that has not received much attention from researchers but needs to be explored in Digital Government Solutions, especially since new legal regulations are being put in force in many countries.

1 | Introduction

The availability of realistic test data is of paramount importance for delivering high-quality software, yet the availability of such data remains a challenge in practice. Inadequate or unrealistic test data has less power to uncover defects. The availability of effective test data is particularly difficult to achieve in domains such as Digital Government Solutions (DGS), where regulations

Abbreviations: DGS, digital government solutions; GDPR, general data protection regulation; QA, quality assessment; SUT, system under test.

strictly prohibit the use of real-life raw data for testing purposes. DGS, also referred to as e-Government solutions, are designed to provide public services without using extensive manpower and bureaucracy. These services cover a wide variety of applications such as taxes, utility bills, licences and permits, medical information, post service for official documentation, and so on. They enable the general public to communicate with the government conveniently and efficiently, and are implemented and used in most countries in the world on a smaller or larger scale.

However, there are limitations when it comes to using the real-life raw data that is processed by government entities for activities that are not part of the actual e-Government service provision, such as pre-production testing. In Europe, a large part of these real-life raw data is considered personal data according to the General Data Protection Regulation (GDPR) (European Union 2016). The United States follows a sectoral approach to data privacy protection (Boyne 2018), and the growth of new digital industries has motivated Asian countries, including China, to work on their legislation to restrict the use of personal data (Junke and Tang 2021). These regulations make real-life raw data unavailable for testing. This forces Quality Assurance Specialists all over the world to find solutions for creating or obtaining privacy-preserving test data that is as similar as possible to the real-life raw data processed by the government.

A variety of methods and tools exist for creating privacy-preserving test data. One practical solution is data anonymization, which transforms the real-life raw data by applying some operations on it to effectively remove personal data without degrading the anonymous data utility (Majeed and Lee 2021). Nevertheless, there remains the risk of someone reversing the anonymization algorithm and retrieving personal data. Another one of the many possible options would be to use one of the various existing machine learning models and generate fully synthetic test data that is very similar to the real-life raw data, provided that the model is trained well. A common prerequisite for using most of these well-known options is that they require direct access to the real-life raw data that is used as input for data anonymization or synthetic test data generation. However, failure to gain access to real-life raw data excludes the possibility of using the above-mentioned methods. Even if access to real-life raw data is granted, it raises security concerns as personal data is prone to cyberattacks and other data-related breaches (ENISA 2023).

Another aspect that needs to be considered when creating privacy-preserving test data is the fact that real-life raw data is constantly evolving. Although historical data from months or even years ago, which contains events with timestamps, as well as consistent relations between data subjects, is sometimes important for providing e-Government services, the majority of applications rely on data that reflects the current or recent state of the data subject. For example, banks may query income information about the latest months to calculate credit limits. Some family benefits might be granted only to parents with newborns under a certain age. Some applications may require a recent life event (e.g., birth, marriage, divorce, or death) as input. Therefore, one could say that a static set of test data created for e-Government entities has an 'expiration date', as over time it will become more and more useless for the applications under test. For that reason, it is important that the test data resembling real-life raw data processed by the government is created with knowledge of the evaluation mechanisms of the same real-life raw data and that the test data can be evolved similarly.

One possible solution would be to generate and evolve synthetic test data based on publicly available microdata, open data, or other input that captures essential characteristics and distribution of real-life raw data without revealing any personal information. The objective of this study is to identify and describe existing methods in the field of software testing that can do that.

The rest of this paper is structured as follows. Section 2 summarises the background of this study, including our specific context. Section 3 describes related work. Section 4 provides definitions and explains our research method. In Section 5, the results of our study are presented. Section 6 gives an overview of the limitations of this study, as well as the resulting threats to validity. Section 7 provides the discussion of results, and finally, Section 8 concludes the paper and highlights possible future research directions.

2 | Background

Estonia, one of the leading countries in e-Government development according to the United Nations E-Government survey of 2022 (Affairs and Social 2022), is one of the pioneers in implementing digital government solutions. Estonian e-Government solutions are built on the interoperability framework X-Road.[1] Today, other countries, for example, Finland, Iceland, and the Faroe Islands, have also implemented the X-Road framework, and the first cross-border data exchange project has been started between Finland and Estonia (Jackson et al. 2022).

For several decades, many countries have pursued the decentralisation of government services with the objective of improving service delivery (Gradstein 2017). In decentralised DGS settings, there is no central database that can be queried for all government data. It is a network of government entities that act as data providers and exchange data with other parties (see Figure 1). Driven by the principle 'Data resides where it is created', Estonian e-Government falls into the category of decentralised e-Government. In Estonia, government institutions can be queried via data services that run on the local implementation of the X-Road technology for data that the government stores (Veldre 2016).

Interoperability is the basis of decentralised DGS. Important factors for designing an underlying interoperability framework for e-Government services are well studied (Flak and Solli-Saether 2012; Scholl and Klischewski 2007). The challenges related to testing the applications that use data from data services that run on an interoperability framework as input in decentralised e-Government settings have, on the other hand, received little attention from researchers and industry.

When real-life raw data is used for privacy-preserving test data generation in the case of centralised DGS, there would only be one central government database to consider. However, for decentralised DGS this would mean having access to the real-life

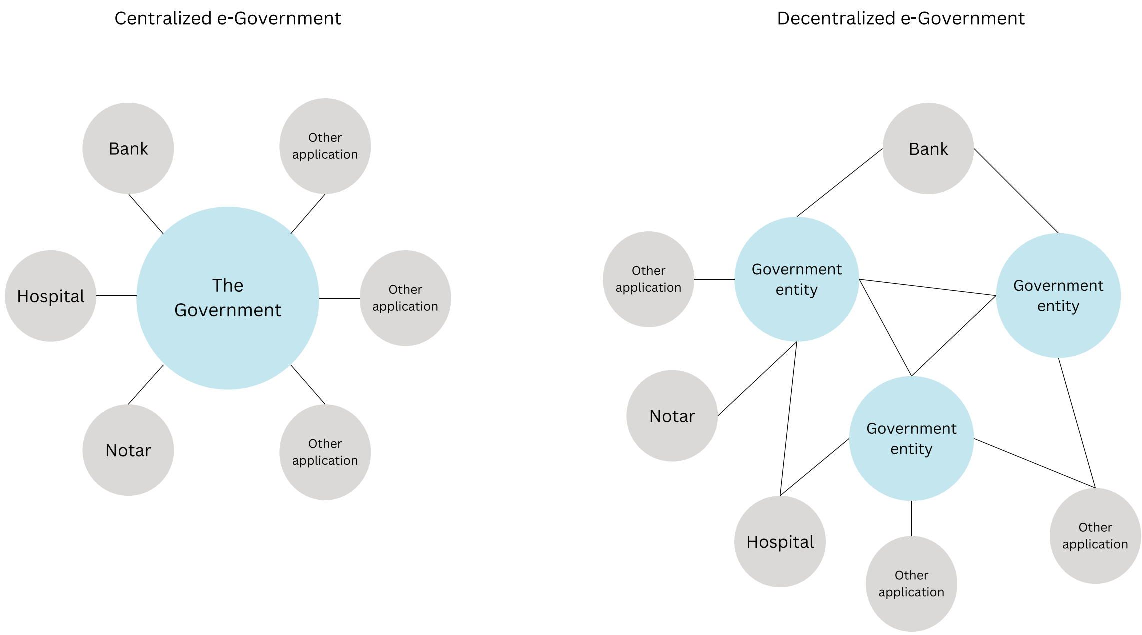

FIGURE 1 | Centralised vs. decentralised DGS.

raw data of all individual government entities. It is a challenge that requires compliance with the individual security procedures of every single government entity. Therefore avoiding directly accessing real-life raw data processed by government entities when creating test data would reduce the risk of data breach as well as the procedural complexity significantly.

From the perspective of the parties who query real-life raw data from government institutions for their systems and applications in production, the data received is thereafter used as input by themselves in their own digitalized work processes, such as providing (public) services, monitoring, reporting, or similar. The increasing number of new systems and applications that are developed and require input test data from government institutions to execute their test cases in pre-production poses challenges as these applications need to be tested thoroughly without the actual real-life raw data being available for testing due to data privacy restrictions.

In decentralised DGS, test data for pre-production testing should be created in a way that the test data of one government entity is compatible with the test data of every other government entity that is part of the same e-Government solution. More specifically, certain identificators must be preserved, and ideally, from all data services offered by government entities in a decentralised e-Government solution, as many as possible should be covered. This is crucial for ensuring that all test data in decentralised DGS are consistent and an instance of test data provided by one government entity (e.g., a test person) has a valid and meaningful match among the test data of another government entity (see Figure 2).

The growing number of projects where proactive services are developed indicates that the test dataset cannot be a static one. Proactive digital services often rely on certain life events that may also be received from data services. Such events may be used as triggers for proactively offering services and benefits to clients, citizens, or residents instead of them applying for these services and benefits. Therefore, any e-Government test data used as input for testing proactive services should have an evolution logic similar to real-life raw data used in production.

3 | Related Work

Due to the need to protect personal data and, in some cases, also the lack of real-life raw data, there is an active area of research that seeks effective methods for generating synthetic data. It is not only the discipline of software testing that requires data that closely resemble real-life raw data, but where all personal data that may lead to the identification of an individual is removed; this includes domains such as public health, digital forensics, finance and banking, government and public services (Tozluoğlu et al. 2023), retail and e-commerce, urban planning, social sciences, and so on. As a result, there are demographic data population datasets that have been created as part of governmental initiatives. Different approaches are used for generating these datasets. For example, statistical approaches are used by the Urban Institute (Pickens et al. 2023).[2] Another example of a statistical-based approach is described in SIPHER Synthetic Population, a dataset that provides a digital twin of the adult population to analyse the impacts of proposed policy changes.[3] Regarding statistical approaches, de Mooij et al. (2024) present a method and tool for creating synthetic demographic populations without using detailed samples but using distributions of aggregated data that reflect spatial, multivariable, and household distributions. The authors do not consider the evolution of individual data over time. Soltana et al. (2017) improved existing usage profiles that mainly focused on embedded and web-based system modelling by using state-machine-like notations (e.g., Markov chains). Their work addresses systems where behaviour is driven by complex, interdependent data that is subject to complex logical constraints, advocating for

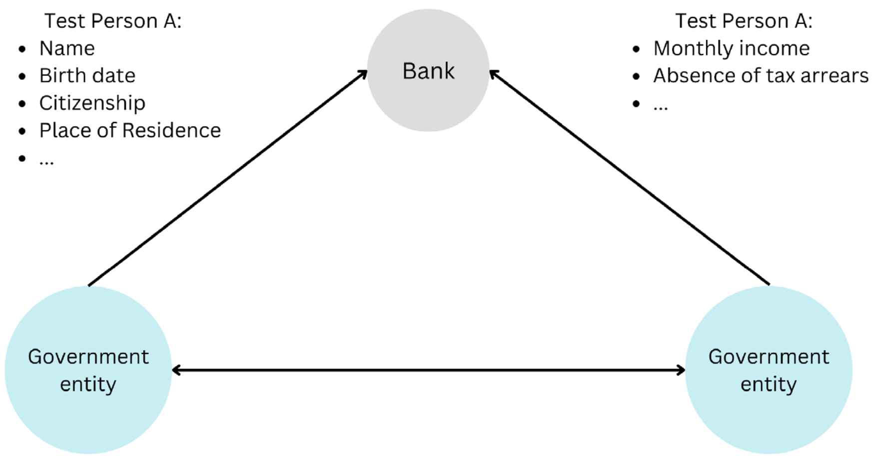

FIGURE 2 | Compatibility of test data among different government entities in a decentralised DGS.

an enhanced data schema with probabilistic information and constraints as a more natural fit for encoding usage profiles. Again, the evolution of synthetic data is not considered. In the work by Prédhumeau and Manley (2023), the authors do consider evolution, and they also use only publicly available real data as the foundation for generating their synthetic population dataset for Canada based on census data and population projections. The authors preserve the privacy by applying several techniques (even though they only use publicly available datasets), however, the evolution of concrete individuals is not considered as the synthetic population is generated independently for each year (2016, 2021, 2023, and 2030).

Another very relevant area is related to health demographics, where data can follow (i) a tabular structure, (ii) is synthetically generated text from medical records, or (iii) it is related to image generation in the health domain. It is particularly relevant to the work around the generation of Electronic Health Records (EHR) where several tools are available, for example, helping to model the spread of infectious diseases and evaluate the effectiveness of interventions. Ash (2017), as part of his doctoral dissertation provides an open-source toolkit to perform synthetic generation and de-identification of human person demographics in the health domain in tabular form. However the evolution is not really considered. Another example is the EHR-Safe framework by Yoon et al. (2023). It produces EHR data using generative adversarial networks (GANs), ensuring both high-fidelity and privacy-preserving synthetic data.

Another possible area of application is related to Machine Learning (ML), for testing applications or models. One domain in which this is very relevant is related to fairness in machine learning Rabonato and Berton (2024). Traditional fairness approaches require demographic data such as race and gender, but these data can be problematic due to inaccuracies or privacy concerns. This area of research would also benefit from approaches that generate synthetic data. Ashurst and Weller (2023) survey also briefly summarises methods to achieve fairness in machine learning without demographic data. Endres et al. (2022) work focused on providing a comparative analysis of different generative models. They observed that most research on generative models has focused on image generation, with an increasing emphasis on text data only in recent years. In a more recent work, Bobadilla and Gutiérrez (2025) describe how Generative Adversarial Networks (GANs) have recently been applied in Recommender Systems by generating augmented data to improve results. Examples include CFGAN and its versions for generating fake purchase vectors, Recurrent Neural Networks (RNN) for temporal patterns in RecGAN, DCFGAN (combined with reinforcement learning), Conditional GANs (CGAN) for conditional rating generation, NCGAN for recommendation training after feature extraction using neural networks, and Hybrid GANs.

The development of ML algorithms relies on training data and is challenged by data privacy requirements, as stated by Abufadda and Mansour (2021), who have reviewed a number of models and studies that proposed generating or using synthetic data for ML in various medical, scientific, and social fields. This study lists several synthetic data generation approaches, but provides only a few insights into each approach listed. There is no evidence that the approaches they identified are able to generate and evolve synthetic data without using real-life data as input. They did not analyse the current trend of Large Language Models (LLMs) for this task, nor did we find relevant approaches to this technology in our domain, which we intend to explore as it is discussed later in this review. From this survey, the authors highlighted some works about evolution, such as the work by Ouyang et al. (2018), which studied the generation of synthetic realistic human location trajectories considering privacy. Although this domain could be interesting for some government entities, we are more interested in life events for data evolution.

The survey carried out by Eigenschink et al. (2023) evaluates deep generative models for synthetic sequential data based on their representativeness, novelty, realism, diversity, and coherence. The authors concentrate on assessing the similarity of the generated synthetic data to real-life raw data, which is also very important in our context. However, another aspect that is relevant to our context, the total avoidance of using real-life raw data, is not considered in this study. Also, although transformer models are discussed, LLMs are not explicitly discussed, so the authors did not find any relevant work about using LLMs for synthetic data generation in 2023. Following this survey, a close match is the work by Lee (2018) in which the authors applied a type of neural network, encoder-decoder model, to generate fully synthetic EHR for clinical decision support and disease surveillance. To generate the model, they used data from around 5.5 million records of emergency department visits, and the authors claim that combined with GANs, such a model can generate comprehensive synthetic EHRs ensuring privacy. LLMs are starting to appear to generate synthetic datasets online, for example, IndoxGen,[4] as well as their evaluation defining metrics about how to compare datasets, measuring the distance between real-life raw data and synthetically generated data.[5]

Also, Göbel et al. (2023) have addressed the large gap between publicly available datasets and actual needs in the field of digital forensics and provided a list of available dataset generation frameworks. Again, the emphasis of this study is not on generating synthetic data without using any real-life raw data in the process and the evolution of the synthetically generated individuals in a realistic way.

Our study fills this gap and concentrates on synthetic data generation approaches that do not use real-life data as input but also considers its evolution as part of the generation of the synthetic dataset. Not only generating multiple datasets at different time intervals, but the individuals that compose the synthetically generated data can be traced.

4 | Research Methodology

Our research team consists of four researchers. We are using Kitchenham's guidelines for performing Systematic Literature Reviews in software engineering for selecting relevant publications (Kitchenham and Charters 2007).

4.1 | Definitions

In the following, we list definitions of three concepts as we understand them in the context of our study.

- Real-life raw data: This refers to data that is created, gathered, and processed in real-world settings and that is not publicly available. In our context, real-life raw data does not include real-life datasets, statistics, or other types of microdata that have been made publicly available. It also does not include the publicly available knowledge or descriptions of real-life raw data, if it is used for synthetic data generation, without having any access to the actual real-life raw data.
- Synthetic test data: This refers to artificially created test data that can be used to replace real-life raw data in high-level system testing.
- Evolving synthetic test data: This refers to transforming the generated synthetic test data over time and in doing so preserving a set of essential attributes of data object instances, for example, the relationships between two or more data subject instances.

4.2 | Research Question

To meet the objective of this study, the research question (RQ) is:

What methods exist for generating and evolving synthetic test data that imitate real-life data without using the respective real-life raw data as input?

The aspects of interest related to our RQ are:

- Type and characteristics of input data used for synthetic test data generation.
- Description of synthetic test data generated.
- Data evolution ability of synthetic test data generation methods.

We are particularly interested in exploring test data generation methods that do not require real-life raw data, yet can generate synthetic test data that closely resembles real-life raw data. In addition, we aim to generate synthetic test data that evolves in a manner closely resembling the evolution of real-life raw data. The evolution of synthetic data can thereby be achieved through different approaches, depending on the context, goal, or research question. Time-series forecasting of movements or events related to one specific entity allows this entity to evolve over time (Gohari et al. 2024). Population-based evolution mainly relies on evolutionary algorithms to evolve a population (Chaudhary et al. 2019), but the individual entities, as well as the relationships between these entities, are mostly not preserved. We are interested in exploring both of these key evolution types in the context of synthetic data evolution.

4.3 | Search for Publications

Inspired by the guidelines of Kitchenham and Charters (2007), the research question is broken down into individual facets (also considering the PICo criteria[6] for qualitative research used for defining the facets, after which a list of synonyms and alternative spellings is created). Before conducting the actual search, trial research strings are created and tested against a list of already known primary studies. The research question facets that are defined as too restrictive while testing trial research strings were removed from the final research strings and included as conditions in the inclusion and exclusion criteria.

Additionally, the list of synonyms is fine-tuned with the help of `wordfreq` (Speer 2022), a Python library for looking up the frequencies of words in many languages, based on many sources of data. This library defines the most frequent keywords describing the Problem, Phenomenon of interest, and Context used in reference articles and it helps the authors to define and include the most important keywords in the search strings.

The following search strings were used:

- *Search string used on IEEE Xplore*: (Advanced search → Command search (Boolean/Phrase): ('software test*' OR 'software quality' OR 'quality control' OR 'quality assurance') AND ('synthetic data*' OR 'data synthesis' OR 'artificial data' OR 'synthetically generated data' OR 'random data generation'))
- *Search string used on ACM Digital Library*: (Advanced search → The ACM Full-Text collection → Search within

Anywhere): ('software test*' OR 'software quality' OR 'quality control' OR 'quality assurance') AND ('synthetic data*' OR 'data synthesis' OR 'artificial data' OR 'synthetically generated data' OR 'random data generation')

- *Search string used on Scopus*: (Search → Refine search → Subject area: limit to Computer Science): ('software test*' OR 'software quality' OR 'quality control' OR 'quality assurance') AND ('synthetic data*' OR 'data synthesis' OR 'artificial data' OR 'synthetically generated data' OR 'random data generation')

After conducting the database search for RQ, all publications found with the search are immediately exported to the Zotero reference management tool[7] where every search result receives a unique ID. For further analysis, the results are exported from Zotero to a shared spreadsheet database that is used as the main working document by all four researchers.

All duplicates are identified and removed before proceeding with the Title and Abstract Analysis.

TABLE 1 | Exclusion criteria.

ID	Exclusion criteria
E1	Book, book section, or a conference review: Justification: this study aims to define approaches that are described with enough detail and quality that they are published in research papers
E2	Full text is not available. Justification: it is not possible to extract the data necessary for our study from a publication that is not available in full. By 'not available' we mean 'it cannot be accessed under our existing licences, and it cannot also be purchased separately online'
E3	Full text is not available in English. Justification: although translation services and software are available for most languages worldwide, we cannot be certain that all technical details are presented correctly if the authors themselves do not translate the paper

4.4 | Title and Abstract Analysis

Once we had all returned publications stored in Zotero and all duplicates removed, we proceeded with the Title and Abstract Analysis stage of our systematic review to narrow down the initial pool of publications. The Title and Abstract Analysis stage was designed to efficiently filter out publications that did not meet our inclusion criteria while ensuring that potentially relevant publications were retained for in-depth examination. It consisted of two main steps: (1) filtering publications based on predefined Exclusion Criteria and (2) selecting publications for full-text analysis using Inclusion Criteria, both detailed as follows.

- *Filtering of Publications based on Exclusion Criteria*: Exclusion Criteria are applied to every unique publication found with the search. The purpose of applying the Exclusion Criteria first is to efficiently exclude publications that cannot be included for Full Text Analysis.
- *Inclusion of Publications for Full Text Analysis*: in the Inclusion for Full Text Analysis step, only the Title and Abstract of publications are read and analysed with regard to our two Inclusion Criteria. To be included for Full Text Analysis, the publication has to be a primary study that meets both of our two Inclusion Criteria.

Every publication included for Full Text Analysis is given a unique ID (P for Publication + number).

The Exclusion and Inclusion Criteria used are provided in Tables 1 and 2.

TABLE 2 | Inclusion criteria.

ID	Inclusion criteria
I1	The publication must mainly suggest and describe an approach or approaches for real-life-like synthetic test data generation Justification: papers that are not mainly concentrated on real-life-like synthetic test data generation are not likely to provide us with enough information to be able to use the approach in our next study
I2	No real-life raw data must be required as input or training data in any step of the test data synthesis process Justification: approaches that use statistics, publicly available metadata, or any other means that do not require direct access to actual real-life raw data are to be defined

4.5 | Full Text Analysis

After completing the Title and Abstract Analysis, we proceeded with the Full Text Analysis stage. It consisted of two main steps: (1) filtering out additional publications that, after reading the full text, had to be excluded based on our Exclusion Criteria or that did not meet our Inclusion Criteria, and (2) extracting data from all included publications, both detailed as follows.

- *Exclusion of Publications*: In case there are secondary studies that were not identified in the previous stage, they will be identified and removed from further analysis. The remaining publications are re-assessed based on our Exclusion and Inclusion criteria to define those where proper exclusion was not possible based on the Title and Abstract only. If a publication gets excluded in this step, the prefix of the unique ID is changed from 'P' to 'Ex'. All publications that are not excluded in this step are selected for further analysis.
- *Data Extraction*: Conceptually, data extraction from the included publications focuses on data extraction items related to

(i) study description, (ii) answers related to our RQ, (iii) quality of the publication, and (iv) maturity of the proposed approach.

The Quality Criteria used for assessing the quality of the publication (iii) are provided in Appendix A.

4.6 | Data Synthesis

To find an appropriate method for synthesising our extracted data, we have looked in the toolbox of Qualitative Data Synthesis (QDS) which is based on identifying common themes across qualitative studies to create a great degree of conceptual development compared with narrative reviews (Hollier 2018). One example of QDS is Thematic Synthesis, a straightforward method with clearly described steps (Flemming and Noyes 2021; Thomas and Harden 2008) where data synthesis is traditionally carried out in several stages. The following stages are used in our study:

1. *Line-by-line coding of the findings of the individual studies*: a mixed approach is used as the data extraction sheet includes some classifications that are defined based on our domain knowledge. Additional coding is done for extracted data items that contain free text.
2. *Development of descriptive themes*: reviewers group the created codes into a hierarchical tree structure based on code similarities and differences.

5 | Results

5.1 | Results of the Search for Publications

Our search across the three selected databases produced a list of 1013 publications. Table 3 shows the number of publications found in each of the digital libraries used.

All publications were imported into Zotero and organised into three separate folders. Among the full list of publications (1013), 45 duplicates were identified and removed from further analysis.

5.2 | Results of the Title and Abstract Analysis

The first step of the Title and Abstract Analysis involved pre-selecting publications and excluding those that did not meet our Exclusion Criteria for Full Text analysis. The following number of publications were excluded:

TABLE 3 | Digital libraries used.

Digital library	URL	# of papers
IEEE Explore	https://ieeexplore.ieee.org/	210
ACM DL	https://dl.acm.org/	402
SCOPUS	https://www.scopus.com/	401

- E1: 127 books or book sections excluded.
- E2: 20 publications were excluded because the full text was not available.
- E3: 2 publications were excluded because the full text was not available in English.

After excluding 149 publications in the pre-selection stage, we continued our analysis with a list of 819 and proceeded with evaluating each publication based on our Inclusion Criteria. To include a publication for Full Text Analysis, it had to be a primary study where both of our two Inclusion Criteria had to be fulfilled.

We quickly discovered that I2 was often difficult to evaluate based on Title and Abstract only, therefore in order to not lose any relevant findings, we decided to include all findings where it was not clearly understandable from the Title and Abstract that real-life data is used as input for the suggested approach.

- I1: 686 publications were not included because it was evident from the Title and Abstract that they do not suggest or describe an approach or approaches for generating synthetic test data that resembles real-life data.
- I2: 58 publications were not included because it was clear from the Title and Abstract that real-life raw data is required as input or training data in the test data synthesis process.

As a result of the Title and Abstract analysis, 75 publications were included for Full Text Analysis and Data Extraction.

5.3 | Results of the Full Text Analysis

After reading the full texts, it was clear that nine of the publications (Ex30, Ex38, Ex39, Ex40, Ex44, Ex45, Ex46, Ex54, Ex63) did not suggest any novel synthetic data generation approach as required in I1, therefore, they were excluded from the final selection. Additionally, another 29 of the included publications suggest a synthetic data generation approach where real-life raw data is used as input for generating synthetic data. Consequently, these 29 papers were also excluded based on I2.

Therefore, 38 publications in total were excluded in the Full Text Analysis stage, leaving 37 publications available for synthesis of the extracted data. The process followed for selecting these 37 publications is illustrated in Figure 3. The representation of the process is inspired by the PRISMA (Page et al. 2021) flow diagram.

The demographics of these 37 selected publications are provided in Appendix B.

5.4 | Answer to the RQ—Results of the Synthesis of Extracted Data

We were prepared for the possibility of not finding a publication that provides us an answer for our whole RQ 'What methods

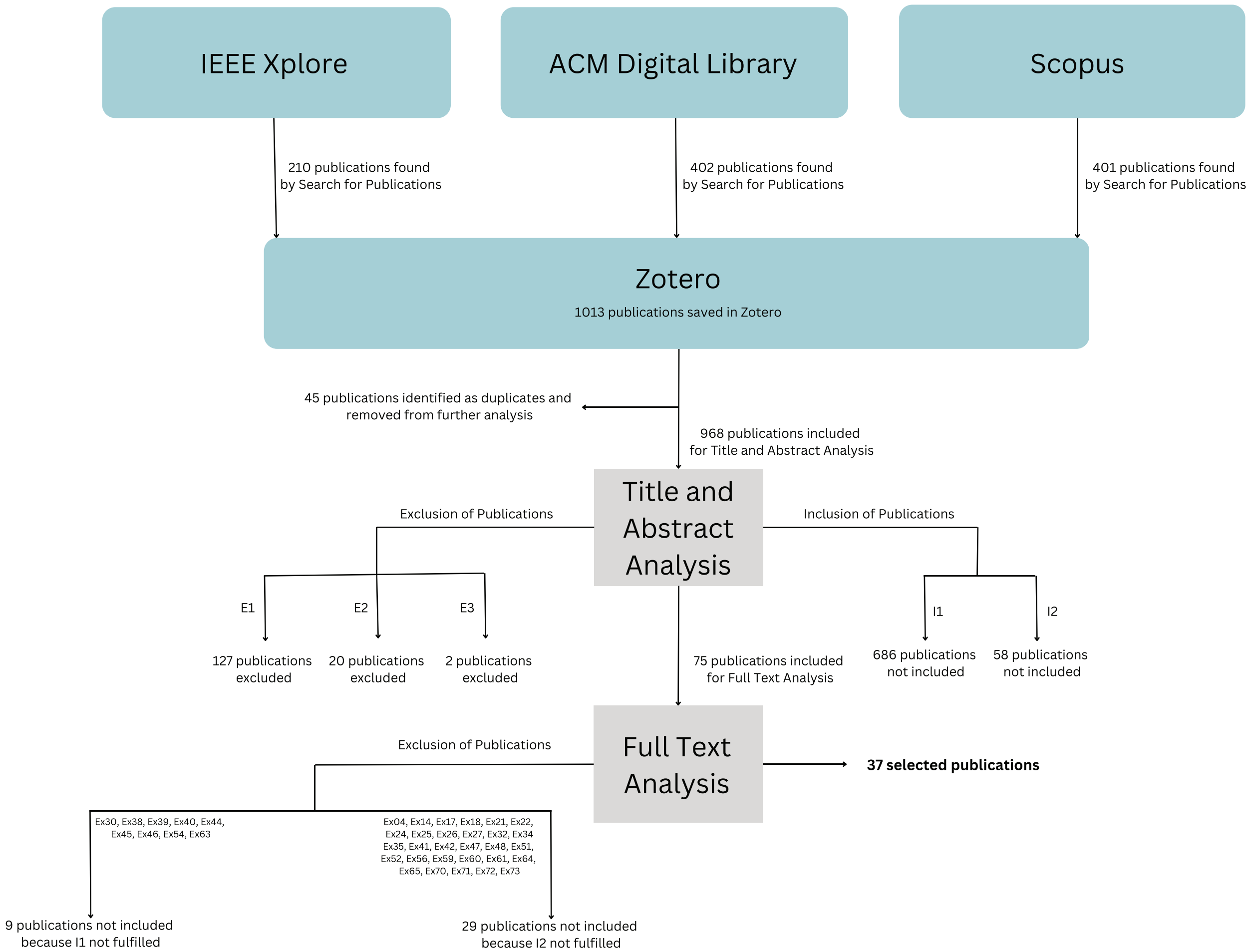

FIGURE 3 | Results of the title and abstract analysis and the full text analysis.

exist for generating and evolving synthetic test data that imitate real-life data without using the respective real-life raw data as input?' Nevertheless, we were hoping that there were existing approaches that answer it partly.

Therefore, we have split our RQ into individual RQ facets. The purpose is to look for answers with each one of these facets and to find out, which are the selected publications that answer the most of them.

5.4.1 | What Methods Exist for Generating Synthetic Test Data?

Our first Inclusion Criterion (I1) was designed as our 'line of defense' for not selecting publications that do not even suggest a novel synthetic data generation approach. Therefore, all of our 37 selected publications provide an answer to the RQ facet 'What methods exist for generating real-life-like synthetic test data?'

We have categorised the different types of approaches presented in the 37 selected studies as follows:

- *Rule-Based generation*: approaches where synthetic test data is generated based on specific user-defined rules, or where the source code or the System Under Test (SUT) is used for defining the rules for synthetic test data generation.
- *Evolutionary Algorithms*: Algorithms that are based on the idea of evolution, for example Genetic Algorithms.
- *Classification/Regression Models*: Algorithms that generate synthetic test data based on previously trained classification/regression models.
- *Deep Learning*: Neural Networks with multiple layers of interconnected nodes, also referred to as neurons or units.
- *Image/Video Rendering Tools*: approaches where synthetic test data is created by using image and/or video rendering tools.
- *Simulation Environments*: approaches where synthetic test data is created by using simulation environments.
- *Other*: types of approaches that were suggested only once in our population of 37 publications.

There were a few cases where a suggested approach seemed to combine more than one method. For example, the approach suggested in P30 was classified as 'Rule-Based generation', but there remains the question if the tool developed by the authors might also use a Search-Based algorithm, as meta-heuristics are used in the induction of the rules. In order to avoid unnecessary complexity in categorising the types of approaches, we looked at this question from the viewpoint of the user of the approach and asked ourselves how the user of this approach would identify it. Would the user need to know the ‘business logic’ to be able to create rules? Or understand how a search problem is solved? Or maybe have to have access to training data to train a model? The answers to these questions helped us identify the most relevant category from the user's point of view.

A total of 13 studies suggested a synthetic test data generation approach based on rules. This includes Web Services Description Language (WSDL) specifications (P01), information from specification/implementation or other formal requirements (P02, P07, P11, P19, P20, P53, P67, P69, P74, P75), tokens and grammar rules (P03), and UML diagrams and OCL constraints (P28).

The category of Evolutionary Algorithms includes five suggested approaches. Genetic Algorithms (P06, P08, P09, P13) e.g., Search-based mutation testing (P05), are used in all cases.

We identified two cases where Classification/Regression Algorithms were used. A white-box regression model was suggested, where the structure of the model is available and can be used for test case generation (P12). Another publication (P62) offered a solution for classifying datasets that have great variability in the number of attributes, types of attributes, and number of class values.

The Deep Learning category includes five approaches where Deep Neural Networks (P29) or GANs (P36, P49, P58, P66) are used for synthetic data generation.

Image/Video Rendering Tools were used in three studies. One publication (P15) used both commercial, and open-source software for synthesising a 3D scene model with a city model and pedestrians and another publication (P55) combined two existing image rendering tools to synthetically generate facial data. In our third approach in this category, CAD models were combined with physical objects (P31) to generate data for manufacturing.

In the Simulation Environments category, we had an approach called SynTiSeD (P37) where simulation environments were used to synthesise energy consumption data. We also identified an approach SoccER (P57) that was built on the existing upon the Gameplay Football simulator and used to simulate football games with synthetic data. Our third selected publication in this category used simulation for creating datasets for the evaluation of Multi-Target Regression and Multi-Label Classification methods (P68).

The types of approaches that occurred only once among our 37 selected publications, including one paper where data obfuscation was used (P23), were assigned to the Other category. There are six approaches in total in this category, and they include:

- A publication (P10) that aimed to evaluate the performance of the hill climbing search algorithm compared to random test data generation in a very specific context.
- A Successive Random Addition (SRA) method for creating synthetic weather patterns (P16).
- A publication where synthetic training data with various types was created to train an aspect-based sentiment analysis model (P23).
- An approach called BackTranScription (BTS) that was suggested by the authors in their previous work and used for synthesising speech in Korean (P33).
- A Domain-Specific Language (DSL) called Steveflex that was designed for creating synthetic data to test data-intensive software systems (P43).
- A generative hierarchical probabilistic dynamic model was proposed that explicitly models large spatial and temporal variations in human motion sequences (P50).

The purpose of generating synthetic test data with the suggested approaches was in roughly half of the selected publications (19) to facilitate software testing at different levels, from unit testing to high-level system testing. Other purposes included e.g., generation of training data for Deep Neural Networks, assessing or controlling the quality of certain domain-specific data, evaluating specific algorithms or models, system development in general, or research.

Specific limitations related to the suggested approaches were mentioned in 17 of the included publications. The limitations were concerning (i) the computational resources required for synthetic data generation, (ii) the performance of the suggested approach or tool, (iii) the limited types of output data that can be generated, and (iv) the quality of synthesised data. There are 20 studies that did not mention any limitations related to the suggested approaches.

5.4.2 | What Methods Exist for Evolving Synthetic Test Data?

In order to answer this facet of our RQ, we first had to clearly define the meaning of ‘evolving synthetic test data’ in our context. A definition is given in the Subsection 4.1, and it is important to note that it refers to the evolution of a set of synthetic data that is already created. This means, that in our context, data evolution cannot be a part of the synthetic data generation process. It is an independent process that can start only once the original synthetic test dataset is fully generated and its quality verified. Based on these thoughts, we were able to rule out Evolutionary Algorithms that were used for synthetic data generation as well as any other means of evolution that were not designed as an ongoing process for constantly evolving the synthetic test data.

Among our 37 selected publications, we identified only two studies that provided an answer to our RQ-facet ‘What methods exist for evolving synthetic test data?’ (see Figure 6). Both studies are a part of the same research at the University of Oslo, Norway, in cooperation with Testify AS. The context of this research is in fact very similar to ours, as it was aimed at finding a solution for synthesising and evolving test data that imitates the actual data in the Norwegian National Registry.

The first one of the two publications (P66) was published in 2019 and suggests Multi-layer Recurrent Neural Networks for simulating life events (e.g., births, marriages, deaths) that are used to evolve the initial synthetic dataset that was created in the same publication. Although not specifically mentioned, it is likely that this approach is able to preserve essential attributes of data object instances. The training dataset that was used in this study was collected from a test environment of the Norwegian National Registry. The training dataset itself was therefore synthetic and generated in the course of several years either manually or through the execution of automated test suites. As a result, the statistical characteristics of this dataset were very different from those of the real Norwegian National Registry data.

The same authors have suggested a new approach (P43) in 2023 where a DSL model was used for synthetic data generation. The DSL model was retrained on a quarterly basis on a training corpus that was composed of 100 days of production data that they had access to. Retraining the model on actual production data and regeneration of the whole synthetic dataset allowed to evolve the generated synthetic data on a quarterly basis. From the paper, it is not clear to us if essential attributes of data object instances were preserved, or if this evolution approach would allow them to be preserved.

5.4.3 | What Methods Exist for Generating and/or Evolving Synthetic Test Data That Imitate Real-Life Data?

Answering this RQ-facet was challenging, as there were not many of the selected publications that included a thorough description of the synthetic data that was generated. Based on the data extracted from our selected publications, we defined two categories:

- *Real-life like data*: the publication included information about the generated synthetic data being real-life like.
- *NA*: the publication included no or not enough information about the generated synthetic data being real-life like.

As shown in Figure 4, there were 24 studies among our 37 selected publications where we could identify that the generated synthetic data imitated real-life data. In this context, it is important to note that we were not in the position to evaluate the synthetic data ourselves and this decision was therefore made solely based on the information provided by the authors of the studies. In 13 studies, it was unclear if synthetic data was actually produced or if it was real-life like. We found that the most comprehensive descriptions of the synthetic data were in the studies where synthetic or semi-synthetic images were generated. These descriptions were created during the validation or evaluation of the approach.

The two categories of generated synthetic data were distributed among the types of approaches shown in Figure 4.

5.4.4 | What Methods Exist for Generating and/or Evolving Synthetic Test Data Without Using Real-Life Raw Data as Input?

In the Title and Abstract Analysis and Full Text Analysis stages, we excluded all publications that suggested synthetic data generation approaches where real-life data is needed as input. In order to get a better understanding of what our 37 selected publications require as input when generating and/or evolving

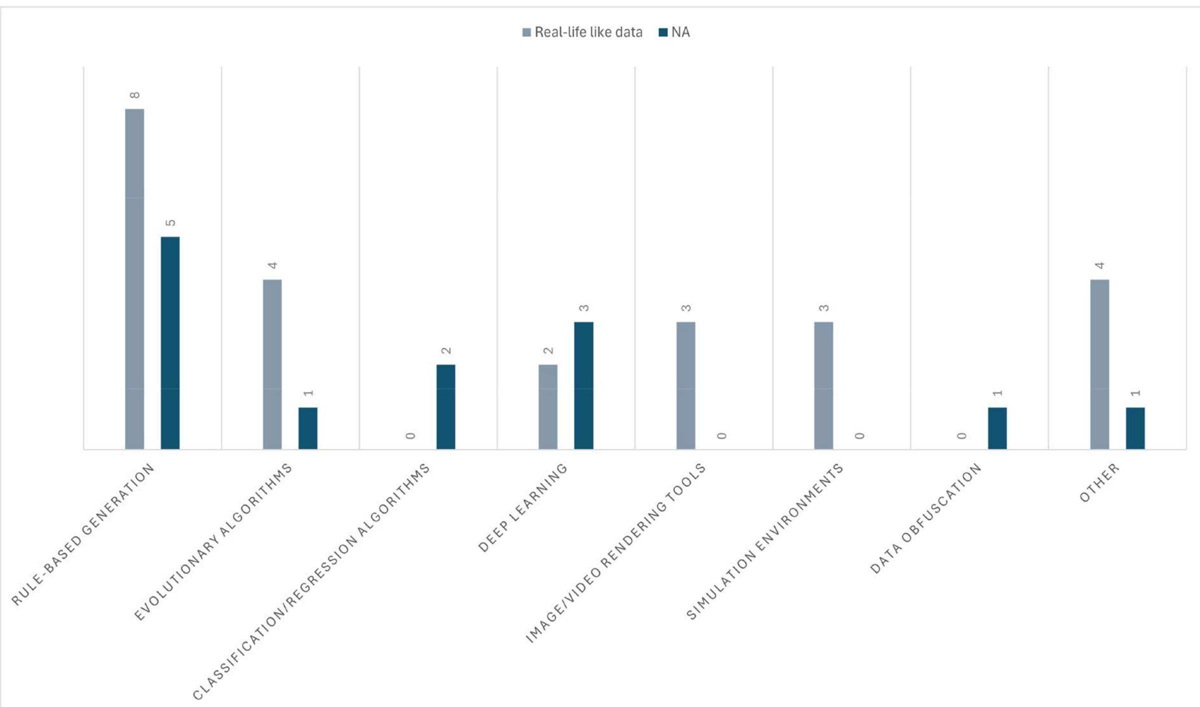

FIGURE 4 | Type of approach—output data.

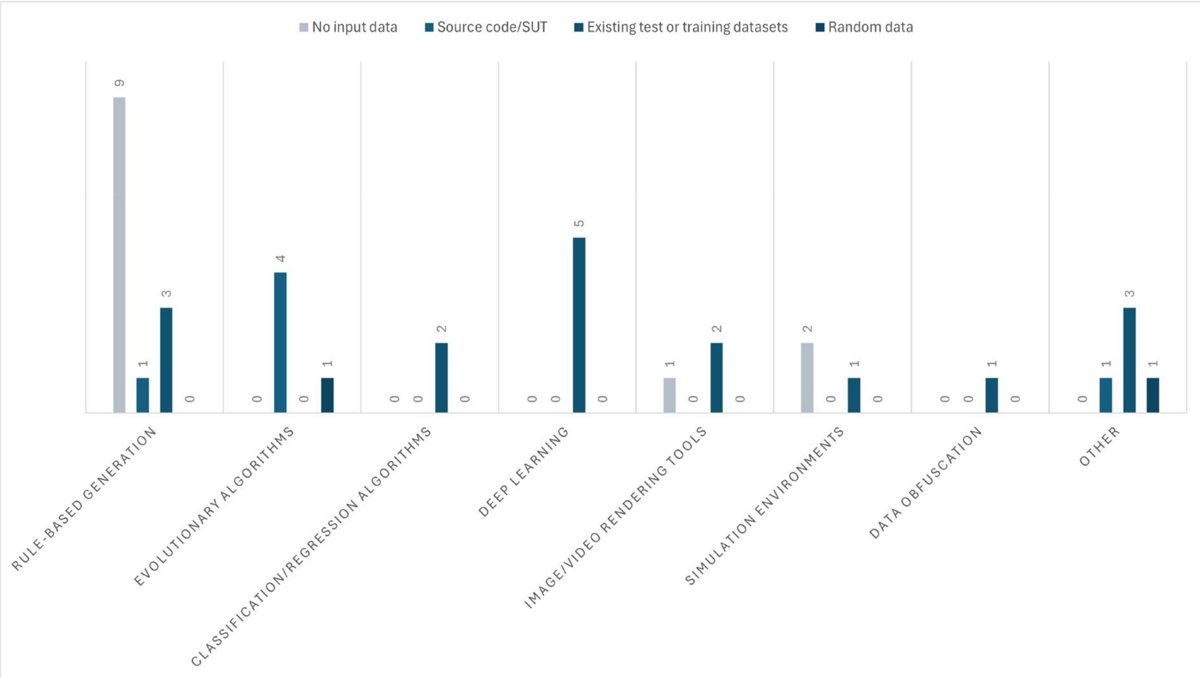

FIGURE 5 | Type of approach—input data.

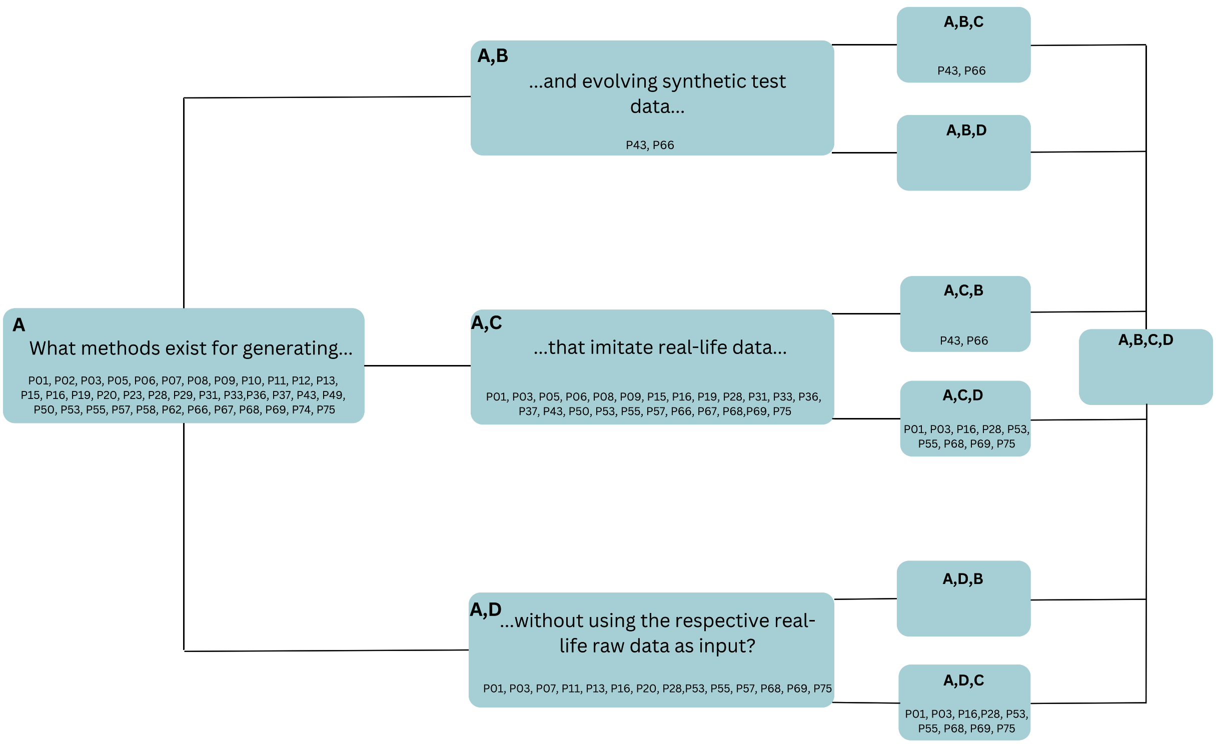

FIGURE 6 | Answers to RQ-facets.

synthetic data, we classified the data item ‘input data that is used as a starting point’ using the following categories:

- *No input data*: synthetic data is generated either according to specific rules defined by the user or with the help of any other means that do not require access to domain-specific data or source code.
- *Source code/SUT*: access to the source code of the SUT is required.
- *Existing test-or training datasets*: access to already existing test data of the SUT is required or publicly available test or training data is used, for example the Iris dataset from the UCI Machine Learning Repository (Fisher 1988).

- *Random data*: data of pre-defined data type (e.g., string, integer) is created randomly without access to domain-specific data or source code.

It is important to note that although no real-life data was used in the studies that relied on publicly available test or training datasets when validating or evaluating their approaches, it is not possible to apply them to real-life systems without gaining access to real-life data. Therefore, these approaches cannot be considered as an answer to the RQ-facet 'What methods exist for generating and/or evolving synthetic test data without using real-life data as input?'

The same applies to approaches that require access to the actual source code of the SUT. In our context, having access to the source code of the SUT cannot define the data that is coming in from external e-Government services. Therefore, in order to synthesise incoming data, these approaches would need access to real-life data, or they would need to be combined with another approach.

Based on these two considerations, our RQ-facet 'What methods exist for generating and/or evolving synthetic test data without using real-life data as input?' can only be answered with selected publications where either no input data or random data is required as input for the suggested data synthesis approaches.

As shown in Figure 5, this is an important limitation that almost entirely excludes the possibility of using Machine Learning approaches that have proven to be very effective for generating large amounts of realistic test data. The limitation is to some extent also relevant to using Large Language Models (LLMs). Therefore, among our 37 selected publications, we were able to identify only 14 approaches that provide an answer for the RQ-facet 'What methods exist for generating and/or evolving synthetic test data without using real-life data as input?'

5.4.5 | What Methods Exist for Generating and Evolving Synthetic Test Data That Imitate Specific Real-Life Data Without Using the Respective Real-Life Data as Input?

There was no approach suggested among our selected publications that provided an answer to all our four RQ-facets. There were however nine publications that came close by answering three out of our four RQ-facets, only missing one single solution for data evolution (see Figure 6).

The majority of the nine publications suggested a Rule-based Generation approach that required no input data (P01, P03, P28, P53, P69 and P75). One approach was based on the Successive Random Addition (SRA) method (P16) and the final two used Image/video rendering tools (P55) and Simulation environments (P68).

Four out of the nine publications stood out in our Quality Assessment where both (P28) or at least one of the researchers (P53, P55, P69) decided that all of our four Quality Assessment Criteria can be graded as 'Yes.'

Two out of the nine publications, P01 and P68, received poor evaluations. P01 was marked with a 'No' for every Quality Assessment Criteria by one of the researchers, and P68 was commented on as a 'very poor paper' by a researcher. The first publication (P01) proposed a testing technique that integrated an external test-case generator into a Property-Based Testing (PBT) tool in order to combine the features of two test-case generation strategies. Rules derived from WSDL descriptions are used as input for generating synthetic test data. The second publication with Quality Assessment results below average (2/4 from Researcher 1, and 1/4 from Researcher 2) and poor researcher comments (P68) was the only one in our 37 selected publications that was not published in peer-reviewed conferences or journals. It is a short paper consisting of 12pages including References and Appendices, and it aims to use simulation for generating synthetic datasets with desired properties (number of examples, data changes events) for the evaluation of Multi-Target Regression and Multi-Label Classification methods.

All nine studies showed at least some validation and/or evaluation efforts with regard to the suggested approaches. There were publications where it was clear to the researchers if and how the validation and evaluation was done (e.g., P28, P53) as well as those where this information was presented rather vaguely (P68, P75).

Six out of the nine studies used tool support (P01, P03, P28, P53, P55, P69) when generating synthetic data, and for two studies the tool was accessible for the researchers at the time of this review (P28, P53).

- *P01*: Rule-based generation of test cases. Might be usable to some extent when generating synthetic data based on WSDL descriptions. Limitations of the approach were described, and they were related to computational resources required. Poor Quality Assessment results from one Researcher (3/4 Researcher 1, 0/4 Researcher 2). Tool support was used when generating synthetic test data (MoMuT[8] and FsCheck[9]), although the tools are available but there was no proper description of its replicability (conference paper).
- *P03*: An integration of grammar-based testing in a framework for contract-based testing in PHP. No limitations described. Quality Assessment results 3/4 (Researcher 1), 1/4 (Researcher 2). Tool support was used (Praspel), but the website that was referenced in the publication as the location of the tool did not include it. Also, the tool does not seem to be active.
- *P16*: The Successive Random Addition (SRA) (Liu et al. 2004) was used for synthetic data generation with the purpose of assessing spatio-temporal data quality. Quality Assessment results were average (2/4). No limitations and no tool support was described.
- *P28*: This publication suggested an approach based on Unified Modelling Language (UML) and OCL (Object

Constrain Language) constraint solving that can generate synthetic data for system testing. OCL constrains solvers, an important part of model-driven engineering (MDE), allow us to find solutions to constraints expressed in OCL. The publication received the highest possible Quality Assessment score from both Researchers (4/4). Limitations were described, and they are imposed by the constraint solver that might not always be exhaustive and find a solution. Tool support was provided, and the PLEDGE (PracticaL and Efficient Data GEnerator for UML)[10] tool was available on GitHub but it seems to be not active at the time of this review.

- *P53*: This publication proposed a new synthetic data generator that was able to generate three-way datasets with planted triclusters (where values are correlated across the three dimensions (observations times features times contexts)) where the user could define several properties regarding the dataset and the planted solutions. The publication received maximal Quality Assessment results from one Researcher, 2/4 (Researcher 1) and 4/4 (Researcher 2). Limitations were discussed, and they are related to computational resources required for synthesising data with this approach. The publication proposed tool support G-Tric[11] and the tool was accessible at the time of this review.
- *P55*: In this publication, two existing tools were combined to synthetically generate facial data. The publication received maximal Quality Assessment results from one Researcher, 2/4 (Researcher 1) and 4/4 (Researcher 2). No limitations were described. Existing image rendering tools (iClone, Blender 3D) were used for synthetic data generation.
- *P68*: This publication uses simulation for generating synthetic datasets with desired properties (number of examples, data changes events) for the evaluation of Multi-Target Regression and Multi-Label Classification methods. Quality Assessment results are below average (2/4 Researcher 1, 1/4 Researcher 2). The limitation of this approach is the limited number of inputs and outputs. Tool support is not described.
- *P69*: This paper focuses on automatically generating valid test input data for jUnit tests based on the provided Design by Contract (Meyer 1997) specification and with the help of mocking. The publication received half the marks in Quality Assessment results from one researcher and maximal from the other (2/4 Researcher 1 and 4/4 Researcher 2). Limitations are related to the performance of the suggested approach and the quality of generated synthetic data. Tool support is not described.
- *P75*: This publication describes a Rule-based software test data generation approach that is proposed as an alternative to either path/predicate analysis or random data generation. It was the oldest publication that was found with our search strings, as it was published in 1991. The publication received a Quality Assessment score of 3/4. No limitations and no tool support were described in this publication.

We publicly provide the spreadsheets used for the analysis of both steps, that is, Title and Abstract Analysis worksheet as well as the data extraction sheet used for the Full Text Analysis on Figshare.[12]

6 | Limitations and Threats to Validity

Building on the reflections presented in the work of Verdecchia et al. (2023), Lago et al. (2024), we systematically identified the limitations of our study and the associated Threats to Validity (TTV), as well as the causal relationships between these two aspects. In our context, limitations refer to the inherent constraints of the study's scope and design, whereas TTV represent the potential consequences of these constraints on the credibility and generalizability of our findings. By explicitly analysing these relationships, we aim to provide transparency regarding the robustness of our methodology and the reliability of our conclusions.

From the variety of existing types of threats, we have defined and discussed those that are relevant to our research method. They are defined as follows:

- *Internal Validity* examines whether an experimental treatment/condition makes a difference, and whether there is evidence to support a claim.
- *External Validity* concerns itself with whether the results can be generalised (Ampatzoglou et al. 2019).

The list of Limitations and the resulting TTV together with appropriate mitigation strategies (where applicable) is provided in Tables 4 and 5. The column 'Conclusion' in Table 5 states if a TTV was accepted as it is or if actions were taken to reduce the effect of a TTV.

7 | Discussion

As stated previously, our study aims to identify existing synthetic test data generation approaches that can be used in real-life context without having access to real-life raw data. In addition, we are interested in the ability of these approaches to evolve the generated synthetic data. Considering that our research question "What methods exist for generating and evolving synthetic test data that imitate real-life data without using the respective real-life raw data as input?" is quite restrictive, it is not surprising that our set of selected publications includes only 37 studies. Considering that software testing is not a new discipline and the need for test data has been there for decades, it is interesting that the majority of our selected publications (25) were published after 2015. It might be related to the fact that we were specifically looking for approaches where real-life raw data is not used in any step of the process. It is also quite recently that following the GDPR (European Union 2016) that dates back to 2016 and entered into force in 2018, many countries are enforcing stricter personal data protection laws. This has made access to real-life raw data increasingly complicated in the field of software testing.

On the other hand, it shows that it is important to find efficient methods for creating fully synthetic test data without having access to actual real-life raw data.

TABLE 4 | Limitations and threats to validity.

Limitation	Threats to internal validity	Threats to external validity
A limited number of databases were searched in this review	IV1: The list of papers found may not be the full list of papers available in the world that answer our RQ	—
The Exclusion and Inclusion of papers based on Title and Abstract was done mostly by the Principal Researcher	IV2: Exclusion and inclusion results may be based on biased decisions	—
Data extraction is done by Researchers and not automated, therefore it is to some extent subjective	IV3: Data extraction results may be biased and miss important data	—
The keywords describing the Population were not included in the Search String as they made the Search String too restrictive	—	EV1: The results of this review may not be transferrable to other similar Populations
There are no standard criteria and metrics for Quality Assessment in secondary studies	IV4: The choice and usage of Quality Criteria may be arbitrary	—

Two studies among our 37 selected publications suggest a synthetic data generation approach as well as a synthetic data evolution approach. Surprisingly, both studies share three authors from the University of Oslo, Norway, who are working in co-operation with Testify AS for a goal very similar to ours, that is, generating and evolving synthetic data for testing Digital Government Services. The authors start out without using actual real-life data from the Norwegian National Registry in 2019 and suggest Multi-layer Recurrent Neural Networks that are trained on synthetic test data. They also admit that the statistical characteristics of their training data are different from the real-life raw data that is processed by the Norwegian National Registry. Four years later, in 2023, the same authors have switched over to using real-life raw data for training their suggested DSL model. This sequence of events illustrates the complex challenge of generating synthetic data without using actual real-life raw data as input, as well as the importance of creating synthetic data that closely resembles real-life raw data. It also demonstrates that generating and evolving synthetic data for testing e-Government services is an important topic relevant in many countries where these services are widely used. Not only that, considering that many governments are on the digitization path, the number of these countries is likely to increase soon.

Although our strict Inclusion Criteria helped us to select only studies where no real-life raw data is used as input when generating synthetic data, answering our RQ-facet concerning input data forces us to really think about how approaches that are validated or evaluated only with test or training datasets, source code or SUT could be applied in the context where no real-life raw data is available and no access to any source code or SUT is granted for security reasons. These restrictions cover other domains that are different from ours, such as images, but as we also were searching for techniques that can be applied to our domain, we were more open-minded in that respect. The test or training datasets, source code, or SUT that are used for validation and evaluation in these studies cannot help us in real-life context, only real-life data can. Therefore, we consider these types of input data to be equivalent to real-life data in our specific context.

We also realised that although state-of-the-art LLMs are used in synthetic data generation approaches and tools in the Industry, there is still not much relevant peer-reviewed research on synthetic data generation and evolution with the help of LLMs. Our specific context and our strict Exclusion and Inclusion criteria limit the choice from existing LLM-based approaches even more.

Among our 37 selected publications, none fully address our research question RQ. Nevertheless, nine studies do not include synthetic data evolution ability but provide an answer for all three other RQ-facets out of four RQ-facets. Of these nine studies, there are two that stand out as being of similar context to ours, showing great Quality Assessment results and Researcher comments, discussing the limitations of their approaches, and providing tool support (P28, P53). These two approaches are good candidates for future work. The sample input files provided by P28 show this approach has the potential for synthetic data generation in different domains from e-Government services to satellite communication. Although the prototype of the tool is publicly available, the source code is not publicly shared and would need to be retrieved before the approach can be adjusted and implemented for synthetic data generation in the Estonian e-Government settings. P53 allows the creation of symbolic and numeric datasets, but the current approach enables only one type to be chosen per dataset. This is an important limitation when applying this approach to synthetic data generation in Estonian e-Government settings, considering the complexity and variety of required synthetic data. Nevertheless, even when the approach and tool suggested in P53 cannot be directly applied or even developed further, the idea of describing events related across several dimensions (three, in this case) and having properties that evolve with them is interesting and potentially valuable when creating a synthetic data evolution approach.

The publications selected were also applied to different domains such as code or images that are not easily translated into our domain, data related to individuals that evolves over time, and life events. We were hoping that although the domain could differ, the techniques could still be applied.

TABLE 5 | TTV mitigation strategies.

TTV #	Mitigation strategy	Conclusion
IV1	The databases (IEEE Explore, ACM DL, and Scopus) that are the main sources for quality research in the field of Software Testing are selected. They contain papers that have been reviewed and therefore they have passed a preliminary quality inspection	The researchers accept the risk of missing some papers when not searching additional databases and grey literature because it is likely that most of these papers that are not published in journals will not meet the Exclusion-, Inclusion-, Quality-, and Maturity criteria of our study
IV2	To validate the decisions of the Principal Researcher, 20 papers were randomly selected for Abstract analysis and Exclusion/Inclusion of papers by second reviewers The results of the experiment showed that three papers out of 20 were either excluded or not included by the Principal Researcher but they were included by a second reviewer The full text of these three papers were then read by a second reviewer who decided that the stricter Exclusion/Inclusion strategy of the Principal Researcher is justified so that these papers should not be included	The researchers accept the risk of possibly not including up to 15% of publications due to the stricter Exclusion/Inclusion strategy, because the stricter strategy was justified by the experiment
IV3	To validate the data extraction results of the Principal Researcher, data from X papers was extracted by second reviewers The extracted data for the data items E-RQ-1, E-RQ-2, E-RQ-2.6, E-RQ-3, E-RQ-3.1 and E-RQ-4 that were used for defining the relevance of each paper were therein after compared. There were nine principal discrepancies that were cleared, and the suggestions of the Principal Researcher were accepted	The Researchers accept the risk of possibly having biased data extraction results and missing important data because the experiment shows that the risk of failing to define the relevance of papers correctly based on the data extraction results of the Principal Researcher is low
IV4	An approach proposed by Dybå and Dingsøyr (2008) for assessing the quality of Qualitative Research by Principles of Good Practice for conducting Empirical Research in software engineering were customised and implemented	The Researchers have taken action to reduce the effect of this TTV
EV1	Initial testing was done with trial test strings. The results of these tests showed the need to remove the keywords for Population from the search string because they made the search string too restrictive. This RQ facet was therefore included on the Exclusion/Inclusion criteria, as well as in the process of defining the relevance of papers	The Researchers have eliminated this TTV by defining the Population in the Exclusion and Inclusion Criteria

We need a metric(s) that tells us about the quality of the data if the evolution of the metric(s) follows the same value(s). These metrics would need to compare datasets, how similar two datasets are, and using these metrics along the time, if their differences are maintained.

8 | Conclusions and Future Work

Digital government or e-Government solutions are implemented and used in many countries in the world to different degrees. The decentralisation of e-Government solutions, together with strict personal data protection laws, poses new challenges to software testers testing these solutions, as test data received from one e-Government entity must be compatible with test data received from another e-Government entity. Real-life raw data processed by all e-Government entities may not be accessible at all, even for the generation of synthetic data.

We conducted a literature review to identify existing approaches that can be used for generating real-life like synthetic data without using real-life raw data as input. We were also interested in finding out if any of the identified approaches is also able to evolve the generated synthetic data.

We found that although there are many synthetic data generation approaches available, the majority of them require real-life raw data when applied in a real-life context. Even worse, in our case, approaches that generate synthetic data as well as evolve the generated synthetic data over time are very rare. The analysed publications reveal a diverse landscape of synthetic test data generation approaches, categorised into rule-based methods, evolutionary algorithms, classification/regression models, deep learning techniques, including GANs, image/video rendering tools, simulation environments, and unique or hybrid approaches. While these approaches demonstrate the feasibility of generating real-life like synthetic data, most approaches require access to real-life raw data for training or validation, which limits their applicability in privacy-sensitive contexts such as e-Government systems. Only two studies proposed mechanisms for evolving synthetic data over time, and none fully satisfied all four facets of our RQ: generation, evolution, realism, and independence from real-life data. Furthermore, nine approaches stood out by meeting three of the four RQ facets, primarily rule-based or simulation-based methods, but they lacked evolution capabilities. Tool support was reported in several cases (e.g., PLEDGE, G-Tric), though availability and maintenance varied. Common limitations across studies included high computational costs, limited scalability, and restricted output diversity. These findings underscore a significant research gap: the absence of approaches that combine evolution, realism, and privacy compliance without relying on real-life raw data. We were also surprised that generative approaches had not been more widely explored. This gap highlights the need for future research in this direction to develop new methods that can generate and evolve synthetic test data while adhering to strict privacy regulations.

Our future work includes investigating the identified Rule-Based generation approaches more thoroughly with the perspective to enhance them by incorporating LLMs to reduce the heavy workload that is currently required for manually describing the rules and constraints for such approaches. We will also investigate if these approaches could potentially be enhanced with a synthetic data evolution ability. Another direction to explore is the possibility of combining the synthetic data evolution methods identified in P43 and P66 with a synthetic data generation approach that does not use real-life raw data as input. While this study is limited to the field of software testing, cross-domain applications are also to be investigated to identify possible relevant research in other fields of study.

Acknowledgements

The research reported in this article has been partly funded by BMK, BMAW, and the State of Upper Austria in the frame of the SCCH competence center INTEGRATE [(FFG grant no. 892418)] part of the FFG COMET Competence Centers for Excellent Technologies Programme, as well as by the European Regional Development Fund, grant PRG1226 of the Estonian Research Council. Daniel Rodriguez is supported by JCLM project SBPLY/24/180225/000143 under the European Regional Development Fund (FEDER).

Data Availability Statement

The data that support the findings of this study are openly available in Figshare at https://doi.org/10.6084/m9.figshare.25809616.v1.

Endnotes

[1] https://x-road.global/.

[2] https://data.wprdc.org/dataset/synthetic-integrated-services-data.

[3] https://www.gla.ac.uk/research/az/sipher/products/syntheticpopulation/.

[4] https://osllm.ai/2024/10/02/synthetic-data-generation-using-llms/.

[5] https://docs.ragas.io/en/v0.1.21/index.html.

[6] https://utica.libguides.com/c.php?g=960363 $p = 6{,}934{,}097$.

[7] https://www.zotero.org/.

[8] https://momut.org/.

[9] https://fscheck.github.io/FsCheck/.

[10] https://github.com/SNTSVV/PLEDGE.

[11] https://github.com/jplobo1313/G-Tric.

[12] https://doi.org/10.6084/m9.figshare.25809616.v1.

References

Abufadda, M., and K. Mansour. 2021. "A Survey of Synthetic Data Generation for Machine Learning." In *22nd International Arab Conference on Info0rmation Technology* (*ACIT*). IEEE Xplore, 1–7. https://ieeexplore.ieee.org/document/9677302.

Affairs, U.N.D.o.E. & Social. 2022. "E-Government Survey 2022: The Future of Digital Government." https://desapublications.un.org/sites/default/files/publications/2022-09/Report%20without%20annexes.pdf.

Ampatzoglou, A., S. Bibi, M. V. Paris Avgeriou, and A. Chatzigeorgiou. 2019. "Identifying, Categorizing and Mitigating Threats to Validity in Software Engineering Secondary Studies." 106: 201–230. https://doi.org/10.1016/j.infsof.2018.10.006.

Ash, S. M. 2017. "Improving Accuracy of Patient Demographic Matching and Identity Resolution." Master's thesis, University of Memphis. https://umwa.memphis.edu/etd/.

Ashurst, C., and A. Weller. 2023. "Fairness Without Demographic Data: A Survey of Approaches." In *Proceedings of the 3rd ACM Conference on Equity and Access in Algorithms, Mechanisms, and Optimization. EAAMO'23*, New York, NY, USA, Association for Computing Machinery. https://doi.org/10.1145/3617694.3623234.

Bobadilla, J., and A. Gutiérrez. 2025. "Testing Deep Learning Recommender Systems Models on Synthetic Gan-Generated Datasets." *International Journal of Interactive Multimedia and Artificial Intelligence* 9, no. 3: 30. https://doi.org/10.9781/ijimai.2023.10.002.

Boyne, S. M. 2018. "The American Journal of Comparative Law." *American Journal of Comparative Law* 66: 299–343. https://doi.org/10.1093/ajcl/avy016.

Chaudhary, D., A. K. Tailor, V. P. Sharma, and S. Chaturvedi. 2019. "Hygade: Hybrid of Genetic Algorithm and Differential Evolution Algorithm." In *2019 10th International Conference on Computing, Communication and Networking Technologies* (*ICCCNT*), 1–4. https://ieeexplore.ieee.org/document/8944822/.

de Mooij, J., T. Sonnenschein, M. Pellegrino, et al. 2024. "Gensynthpop: Generating a Spatially Explicit Synthetic Population of Individuals and Households From Aggregated Data." *Autonomous Agents and Multi-Agent Systems* 38, no. 2: 48. https://doi.org/10.1007/s10458-024-09680-7.

Dybå, T., and T. Dingsøyr. 2008. "Empirical Studies of Agile Software Development: A Systematic Review." *Information and Software Technology* 50: 833–859. https://doi.org/10.1016/j.infsof.2008.01.006.

Eigenschink, P., S. Vamosi, R. Vamosi, C. Sun, and K. Kalcher. 2023. "Deep Generative Models for Synthetic Data: A Survey." *IEEE Access* 11: 47304–47320. https://doi.org/10.1109/ACCESS.2023.3275134.

Endres, M., A. Mannarapotta Venugopal, and T. S. Tran. 2022. "Synthetic Data Generation: A Comparative Study." In *Proceedings of the 26th International Database Engineered Applications Symposium* (IDEAS'22), New York, NY, USA: Association for Computing Machinery, 94–102. https://doi.org/10.1145/3548785.3548793.

ENISA, T.E.U.A.f.C. 2023. "Enisa Threat Landscape 2023." https://www.enisa.europa.eu/publications/enisa-threat-landscape-2023.

European Union. 2016. "Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the Protection of

Natural Persons With Regard to the Processing of Personal Data and on the Free Movement of Such Data, and Repealing Directive 95/46/ec (General Data Protection Regulation)." https://eur-lex.europa.eu/eli/reg/2016/679/oj.

Fisher, R. 1988. "Iris. UCI Machine Learning Repository." https://doi.org/10.24432/C56C76.

Flak, L. S., and H. Solli-Saether. 2012. "The Shape of Interoperability: Reviewing and Characterizing a Central Area Within Egovernment Research." In *Proceedings of the 2012 45th Hawaii International Conference on System Sciences*. IEEE, 2643–2652. https://ieeexplore.ieee.org/document/6149334/.

Flemming, K., and J. Noyes. 2021. "Qualitative Evidence Synthesis: Where Are We at?" *International Journal of Qualitative Methods* 20. https://doi.org/10.1177/1609406921993276.

Göbel, T., H. Baier, and F. Breitinger. 2023. "Data for Digital Forensics: Why a Discussion on "How Realistic Is Synthetic Data" Is Dispensable." *Digital Threats* 4: 1–18. https://doi.org/10.1145/3609863.

Gohari, H. E., X. H. Dang, S. Y. Shah, and P. Zerfos. 2024. "Modality-Aware Transformer for Financial Time Series Forecasting." In *Proceedings of the 5th ACM International Conference on AI in Finance*, 11. ACM, 677–685. https://doi.org/10.1145/3677052.3698654.

Gradstein, M. 2017. "Government Decentralization as a Commitment in Non-Democracies." *Journal of Comparative Economics* 45, no. 1: 110–118. https://doi.org/10.1016/j.jce.2016.01.005.

Hollier, C. 2018. "Libguides: Good Review Practice: A Researcher Guide to Systematic Review Methodology in the Sciences of Food and Health: 6. Data Synthesis and Summary." https://ifis.libguides.com/systematic_reviews/data-synthesis-and-summary.

Jackson, E. B., R. Dreyling, and I. Pappel. 2022. "A Historical Analysis on Interoperability in Estonian Data Exchange Architecture: Perspectives From the Past and for the Future." In *Proceedings of the 14th International Conference on Theory and Practice of Electronic Governance, 2022*. ACM Digital Library, 111–116. https://dl.acm.org/doi/10.1145/3494193.3494209.

Junke, X., and Y. Tang. 2021. "Legal Protection of Personal Data in China." In: *IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress* (*DASC/PiCom/CBDCom/CyberSciTech*). IEEE, 837–842. https://ieeexplore.ieee.org/document/9730504/.

Kitchenham, B., and S. M. Charters. 2007. "Guidelines for Performing Systematic Literature Reviews in Software Engineering." Keele University and University of Durham, UK. EBSE Technical Report EBSE-2007-01.

Lago, P., P. Runeson, Q. Song, and R. Verdecchia. 2024. "Threats to Validity in Software Engineering – Hypocritical Paper Section or Essential Analysis?" In *Proceedings of the 18th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement. ESEM'24*. New York, NY, USA: Association for Computing Machinery, 314–324. https://doi.org/10.1145/3674805.3686691.

Lee, S. H. 2018. "Natural Language Generation for Electronic Health Records." *NPJ Digital Medicine* 1, no. 1: 63. https://doi.org/10.1038/s41746-018-0070-0.

Liu, H. H., G. S. Bodvarsson, S. Lu, and F. J. Molz. 2004. "A Corrected and Generalized Successive Random Additions Algorithm for Simulating Fractional Levy Motions." *Mathematical Geosciences* 36, no. 4: 361–378.

Majeed, A., and S. Lee. 2021. "Anonymization Techniques for Privacy Preserving Data Publishing: A Comprehensive Survey." *IEEE Access* 9: 8512–8545. https://doi.org/10.1109/ACCESS.2020.3045700.

Meyer, B. 1997. *Object-Oriented Software Construction*. 2nd ed. Prentice-Hall, Inc.

Ouyang, K., R. Shokri, D. S. Rosenblum, and W. Yang. 2018. "A Non-Parametric Generative Model for Human Trajectories." In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence* (*IJCAI-18*), International Joint Conferences on Artificial Intelligence Organization, 3812–3817. https://doi.org/10.24963/ijcai.2018/530.

Page, M. J., J. E. McKenzie, P. M. Bossuyt, et al. 2021. "The PRISMA 2020 Statement: An Updated Guideline for Reporting Systematic Reviews." *British Medical Journal*: n71. https://doi.org/10.1136/bmj.n71.

Pickens, M., J. Andre, and G. Morrison. 2023. "Generating a Fully Synthetic Human Services Dataset: A Technical Report on Synthesis and Evaluation Methodologies." Urban Institute. https://www.urban.org/research/publication/generating-fully-synthetic-human-services-dataset.

Prédhumeau, M., and E. Manley. 2023. "A Synthetic Population for Agent-Based Modelling in Canada." *Scientific Data* 10, no. 1: 148. https://doi.org/10.1038/s41597-023-02030-4.

Rabonato, R. T., and L. Berton. 2024. "A Systematic Review of Fairness in Machine Learning." *AI and Ethics* 5, no. 3: 1943–1954. https://doi.org/10.1007/s43681-024-00577-5.

Scholl, H., and R. Klischewski. 2007. "E-Government Integration and Interoperability: Framing the Research Agenda." 30: 889–920. https://doi.org/10.1080/01900690701402668.

Soltana, G., M. Sabetzadeh, and L. C. Briand. 2017. "Synthetic Data Generation for Statistical Testing." In *Proceedings of the 32nd IEEE/ACM International Conference on Automated Software Engineering* (*ASE'17*). *ASE'17*, IEEE Press, 872–882.

Speer, R. 2022. "rspeer/wordfreq: v3.0. Software." https://zenodo.org/records/7199437.

Thomas, J., and A. Harden. 2008. "Methods for the Thematic Synthesis of Qualitative Research in Systematic Reviews." *BMC Medical Research Methodology* 8: 45. https://doi.org/10.1186/1471-2288-8-45.

Tozluoğlu, Ç., S. Dhamal, S. Yeh, et al. 2023. "A Synthetic Population of Sweden: Datasets of Agents, Households, and Activity-Travel Patterns." *Data in Brief* 48: 109209. https://doi.org/10.1016/j.dib.2023.109209.

Veldre, A. 2016. "Introduction of X-Tee." https://www.ria.ee/en/introduction-x-tee.

Verdecchia, R., E. Engström, P. Lago, P. Runeson, and Q. Song. 2023. "Threats to Validity in Software Engineering Research: A Critical Reflection." *Information and Software Technology* 164: 107329. https://doi.org/10.1016/j.infsof.2023.107329.

Yang, L., H. Zhang, H. Shen, et al. 2021. "Quality Assessment in Systematic Literature Reviews: A Software Engineering Perspective." *Information and Software Technology* 130: 106397. https://doi.org/10.1016/j.infsof.2020.106397.

Yoon, J., M. Mizrahi, N. F. Ghalaty, et al. 2023. "Ehr-Safe: Generating High-Fidelity and Privacy-Preserving Synthetic Electronic Health Records." *NPJ Digital Medicine* 6, no. 1: 141. https://doi.org/10.1038/s41746-023-00888-7.

Appendix A

Quality Assessment

According to Kitchenham and Charters (2007), it is critical to assess the quality of primary studies, in addition to using general inclusion and exclusion criteria. Quality assessment (QA) of studies remains a challenging task despite the variety of available QA instruments and practices (Yang et al. 2021).

For our study, we have implemented a customised version of the quality assessment criteria suggested by Dybå and Dingsøyr (2008). These criteria, listed in Table A1, are proposed for assessing the quality of qualitative research by principles of good practice for conducting empirical research in software engineering. From the original paper, we have excluded the criteria that were not relevant to our study (e.g., criteria that were explicitly aimed at research that involves participants). Quality criteria that were applicable for our study have been customised to the conditions of our research. Each of the criteria must be graded on a 'yes' or 'no' scale, whereby 'no' also represents 'not applicable.'

TABLE A1 | Quality assessment criteria.

ID	Quality assessment criteria
QA-1	Is there a clear statement of the aims of the research?
QA-2	Is there an adequate description of the methodology in which the research was carried out?
QA-3	Is there a clear statement of findings?
QA-4	Is the approach valuable for research or practice?

TABLE A2 | Quality assessment form.

ID	Screening questions	
QA-1	Is there a clear statement of the aims of the research? Consider: 1. Have the authors described the research gap in previous work related to this research (e.g., reference to previous papers of the authors, literature review, lack of related work, etc.)? Is there a rationale for why the study was undertaken?	Yes/No
QA-2	Is there an adequate description of the methodology of the research? Consider: 1. Have the authors described the research methodology in a way that allows the study to be repeated (e.g., detailed process descriptions are described, research data is available, etc.)? 2. Are the limitations of the study discussed explicitly? 3. Is the context in which the research was constructed precise?	Yes/No
QA-3	Is there a clear statement of study outcomes/findings? Consider: 1. Are the findings explicit (e.g., magnitude of effect)? 2. Has an adequate discussion of the evidence, both for and against the researchers arguments, been demonstrated? 3. Has the researcher discussed the credibility of their findings? 4. Are the findings discussed in relation to the original research questions? 5. Are the conclusions justified by the results?	Yes/No

(Continues)

TABLE A2 | (Continued)

ID	Screening questions	
QA-4	Does the study describe the value of the research outcome for research or practice? Consider: 1. Does the researcher discuss the contribution the approach makes to existing knowledge or understanding (e.g., do they consider the findings in relation to current practice or relevant research-based literature)? 2. Does the research identify new areas in which research is necessary? 3. Does the researcher discuss whether or how the findings can be transferred to other populations, or consider other ways in which the research can be used?	Yes/No

The quality assessment form presented in Table A2 was used by researchers of this survey. The quality assessment form includes an odd number of screening questions for every quality assessment criterion. If the majority of screening questions for a criterion are answered with 'yes' then the final grade for the same criterion is 'yes'. Otherwise, the final grade is 'no'.

In this work, the work was divided such that at least two researchers assess every paper according to the quality assessment form. Should they come to a different conclusion regarding a specific criterion, they discuss their assessment results in detail and try to decide on the most appropriate conclusion. Should that not be possible, a third researcher decides the grade of the specific quality criterion.

Appendix B

Demographic Analysis

In the demographic analysis, the 37 selected findings are characterised based on their type of study, geographical location and affiliation of authors. As shown in Figure A1, all selected findings were published between 1991 and 2023, with the majority of them (25) published after 2015.

We have based the geographical distribution of selected publications on the affiliation of the authors. A total of 119 unique authors were

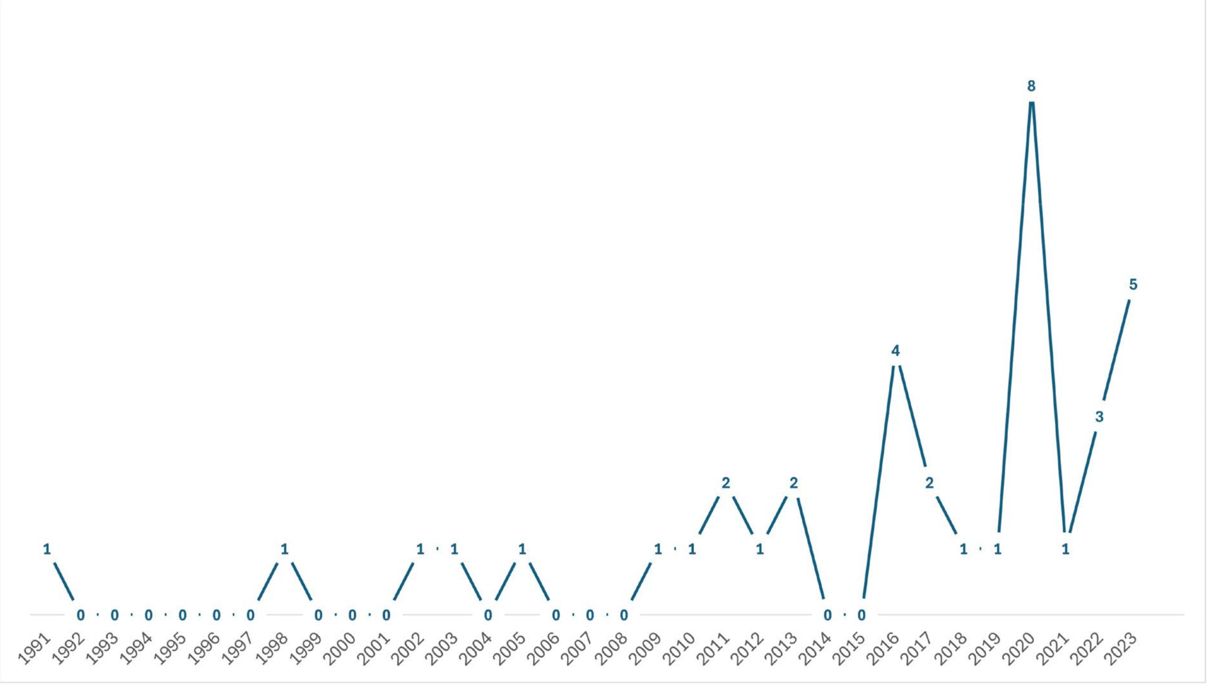

FIGURE A1 | Demographics—timeline.

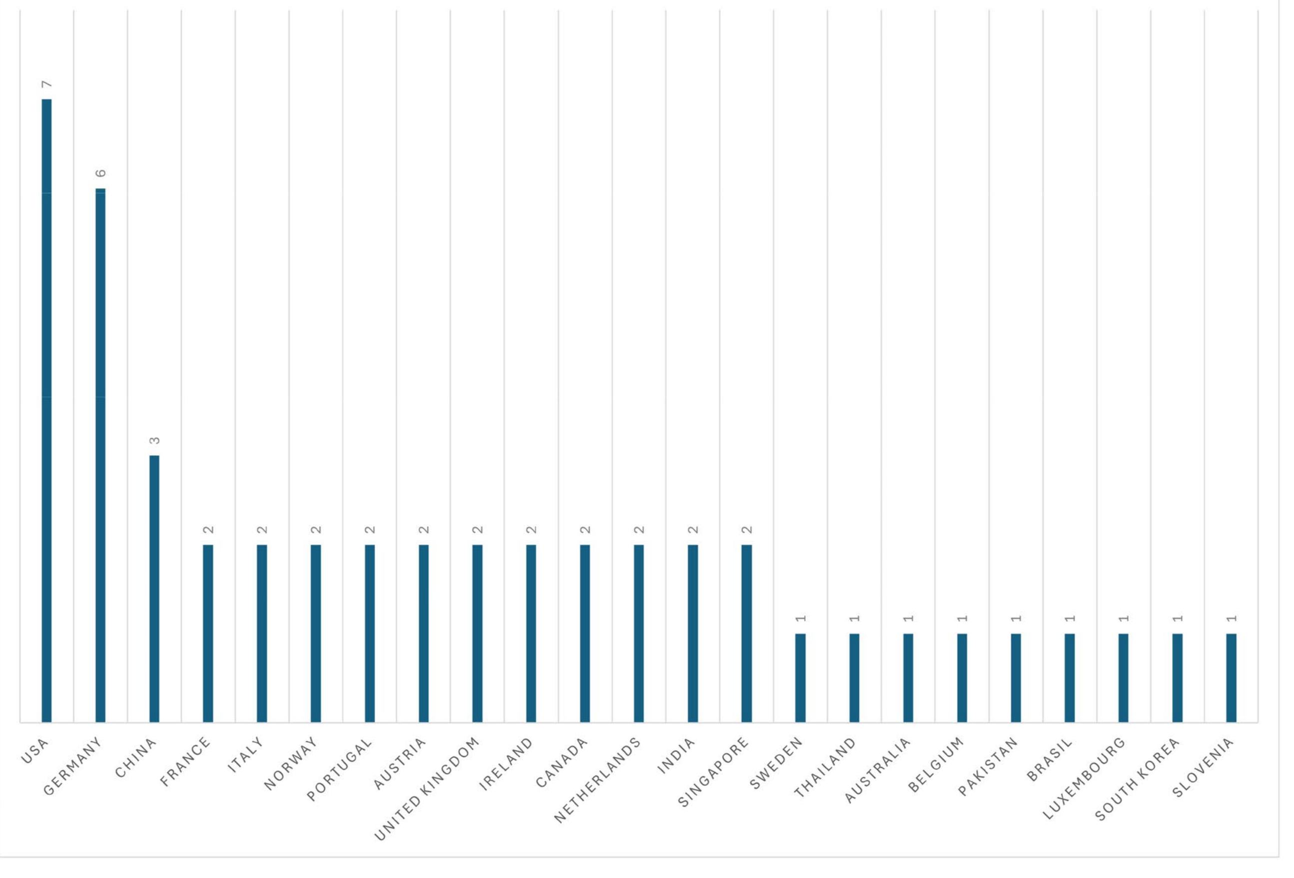

FIGURE A2 | Demographics—geographical location.

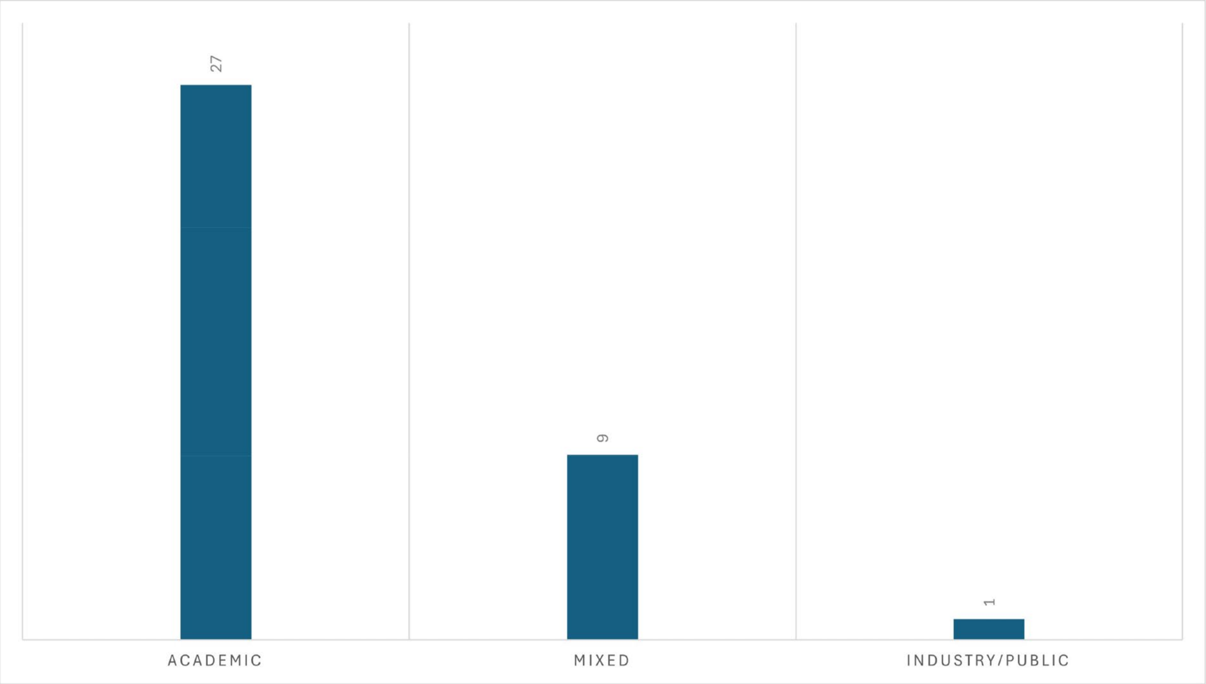

FIGURE A3 | Demographics—type of study.

identified from our selected 37 publications. Three of these authors, Chao Tan (University of Oslo, Oslo, Norway and Testify AS, Oslo, Norway), Razieh Behjati (Testify AS, Oslo, Norway) and Erik Arisholm (Testify AS, Oslo, Norway) were among our list of authors twice with two papers among our selected publications. Our 37 selected publications had authors from 23 countries. The USA and Germany stand out by having affiliations in respectively 7 and 6 of our selected publications. The number of affiliations for each of the 23 countries is shown in Figure A2.

We categorised our selected publications according to the type of study according as follows:

- *Academic*: all authors are affiliated with a university or research institute.
- *Industry/Public*: all authors are affiliated with a company, government institution or state agency.
- *Mixed*: some authors have an academic affiliation and some have an industry/public affiliation.

The majority of our selected publications (27) were purely academic based on the affiliations of authors. There were nine publications with mixed affiliations and one from industry. The distribution by type of study is shown on Figure A3.

Appendix C

Summary of the Literature Search

The summary of literature search results is presented in Table A3.

TABLE A3 | Summary of the number of papers at each stage of the literature search.

Stage	# excluded	# remaining
Total publications retrieved	—	1013
Duplicates removed	45	968
Excluded by Exclusion Criteria (E1–E3)	149	819
Not included by Inclusion Criteria (I1, I2)	744	75
Excluded after Full Text Analysis	38	37
Final selected publications	—	37

Appendix D

Selected Publications

The 37 publications that were selected after Full Text Analysis are listed in Table A4.

TABLE A4 | Selected publications.

ID	Name	DOI or URL
P01	Property-Based Testing with External Test-Case Generators	https://doi.org/10.1109/ICSTW.2017.62
P02	Data coverage testing	https://doi.org/10.1109/APSEC.2002.1183018
P03	Grammar-Based Testing Using Realistic Domains in PHP	https://doi.org/10.1109/ICST.2012.136
P05	An Experimental Tool for Search-Based Mutation Testing	https://doi.org/10.1109/FIT.2018.00013
P06	An Approach for Search Based Testing of Null Pointer Exceptions	https://doi.org/10.1109/ICST.2011.49
P07	Research on Test Automation in the Field of Book Publishing Based on CNMARC Standards	https://doi.org/10.1109/ICIECS.2009.5365826
P08	Evolutionary testing of unstructured programs in the presence of flag problems	https://doi.org/10.1109/APSEC.2005.65
P09	Automatic test data generator: A tool based on search-based techniques	https://doi.org/10.1109/ICRITO.2016.7785020
P10	Search Based Testing of Embedded Systems Implemented in IEC 61131–3: An Industrial Case Study	https://doi.org/10.1109/ICSTW.2013.78
P11	Automatically generating realistic test input from web services	https://doi.org/10.1109/SOSE.2011.6139088
P12	Property-Driven Testing of Black-Box Functions	https://doi.org/10.1145/3524482.3527657
P13	A Strategy for using Genetic Algorithms to Automate Branch and Fault-based Testing	https://doi.org/10.1093/comjnl/41.2.98
P15	DNN Analysis through Synthetic Data Variation	https://doi.org/10.1145/3385958.3430479
P16	A SMART Approach to Quality Assessment of Site-Based Spatio-Temporal Data	https://doi.org/10.1145/2996913.2996932
P19	Systematic Development of Data Mining-Based Data Quality Tools	https://doi.org/10.5555/1315451.1315499
P20	PLATOOL: A Functional Test Generation Tool for Mobile Applications	https://doi.org/10.1145/3422392.3422508
P23	On the Robustness of Aspect-Based Sentiment Analysis: Rethinking Model, Data, and Training	https://doi.org/10.1145/3564281
P28	Practical Constraint Solving for Generating System Test Data	https://doi.org/10.1145/3381032
P29	Fuzz Testing Based Data Augmentation to Improve Robustness of Deep Neural Networks	https://doi.org/10.1145/3377811.3380415
P31	CAD2Render: A Modular Toolkit for GPU-accelerated Photorealistic Synthetic Data Generation for the Manufacturing Industry	https://doi.org/10.1109/WACVW58289.2023.00065
P33	Uncovering the Risks and Drawbacks Associated with the Use of Synthetic Data for Grammatical Error Correction	https://doi.org/10.1109/ACCESS.2023.3310257
P36	GluGAN: Generating Personalised Glucose Time Series Using Generative Adversarial Networks	https://doi.org/10.1109/JBHI.2023.3271615
P37	Generation of meaningful synthetic sensor data — Evaluated with a reliable transferability methodology	https://doi.org/10.1016/j.egyai.2023.100308
P43	Enhancing Synthetic Test Data Generation with Language Models Using a More Expressive Domain-Specific Language	https://doi.org/10.1007/978-3-031-43,240-8_2
P49	Permutation-Invariant Tabular Data Synthesis	https://doi.org/10.1109/BigData55660.2022.10020639
P50	Bayesian adversarial human motion synthesis	https://doi.org/10.1109/CVPR42600.2020.00626
P53	G-Tric: generating three-way synthetic datasets with triclustering solutions	https://doi.org/10.1186/s12859-020-03925-4
P55	Methodology for Building Synthetic Datasets with Virtual Humans	https://doi.org/10.1109/ISSC49989.2020.9180188
P57	SoccER: Computer graphics meets sports analytics for soccer event recognition	https://doi.org/10.1016/j.softx.2020.100612
P58	Medical Time-Series Data Generation Using Generative Adversarial Networks	https://doi.org/10.1007/978-3-030-59,137-3_34

(Continues)

TABLE A4 | (Continued)

ID	Name	DOI or URL
P62	Data Generators for Learning Systems Based on RBF Networks	https://doi.org/10.1109/TNNLS.2015.2429711
P66	Synthetic test data generation using recurrent neural networks: A position paper	https://doi.org/10.1109/RAISE.2019.00012
P67	Synthesis and evaluation of a mobile notification dataset	https://doi.org/10.1145/3099023.3099046
P68	First principle models based dataset generation for multi-target regression and multi-label classification evaluation	https://ceur-ws.org/Vol-2069/STREAMEVOLV3.pdf
P69	Automatically extracting mock object behaviour from Design by Contract specification for test data generation	https://doi.org/10.1145/1808266.1808273
P74	SynConSMutate: Concolic testing of database applications via synthetic data guided by SQL mutants	https://doi.org/10.1109/ITNG.2013.54
P75	A Rule-Based Software Test Data Generator	https://doi.org/10.1109/69.75894

Appendix E

Comparison of Identified Approaches Across Selected Publications

The comparison of the approaches identified across the 37 selected publications is presented in Tables A5–A7.

TABLE A5 | Comparison of synthetic test data generation approaches.

Approach type	# of studies	Key characteristics
Rule-based generation	13	Uses predefined rules or system specifications (e.g., WSDL, contracts, UML/OCL)
Evolutionary Algorithms	5	Genetic/search-based strategies for optimising test data
Classification/Regression Models	2	Trained models used to generate or select inputs
Deep Learning	5	Neural networks (incl. GANs) for synthetic data synthesis
Image/Video Rendering Tools	3	Synthetic images/videos via rendering pipelines or virtual humans
Simulation Environments	3	Domain simulators producing data under controlled settings
Other	6	One-off or hybrid methods (e.g., SRA, DSL-based, probabilistic models)

TABLE A6 | Comparison of approaches.

Type of approach	Publication ID	Domain or purpose	Generated data
Rule-based generation	P01	Model-based testing	Test cases from stateful models
	P02	Data coverage testing	NA
	P03	Unit Testing	Textual data (email addresses)
	P07	Generating test data for the publishing industry	NA
	P11	Testing of web services	NA
	P19	Data scrubbing	10,000 records
	P20	Testing of mobile applications	NA
	P28	System testing	Instance models (several thousand objects)
	P53	Evaluation of triclustering algorithms	Several sample datasets
	P67	Research in the domain of intelligent notification management	32 users, 11,395 notifications, 3148 events
	P69	Unit Testing	EasyMock API calls
	P74	Mutation testing of database applications	NA
	P75	Testing of large software systems	2000 rule-based test cases
Evolutionary Algorithms	P05	Search-based testing of Java programs	NA
	P06	Search-based test data generation	Initial population: 100 individuals
	P08	Evolutionary Testing of Unstructured Programs	NA
	P09	Search-based Testing	NA
	P13	Fault-based testing	NA
Classification/Regression Models	P12	Property-based testing	NA
	P62	Development and testing of data mining algorithms	NA
Deep Learning	P29	Training of Deep Neural Networks	Same type (image) as input data
	P36	Diabetes management	Realistic T1D glucose time series
	P49	Tabular big data synthesis	NA
	P58	Development of data-driven advancements in the healthcare domain	NA
	P66	High-level testing of event-driven systems	6.3 MB of data, 20,844 records
Image/Video Rendering Tools	P15	Analysing limitations of performance of Deep Neural Networks	Urban 3D scene spread across 34.5 km2
	P31	Machine learning algorithms in the manufactoring industry	20,000 images for each tool
	P55	The development of face detection and recognition systems	Several virtual human models
Simulation Environments	P37	Training transferable Non-Intrusive Load Monitoring models	Meaningful synthetic energy datasets
	P57	Automatic event detection	8 complete soccer games
	P68	Multi-Target Regression and Multi-Label Classification	2 datasets, 100,000 examples each
Other	P10	Search Based Testing	NA
	P16	Assessing spatio-temporal data quality	A weather-like phenomenon
	P23	Aspect-based sentiment analysis	NA
	P33	Data quality control	50,000 words
	P43	Testing data-intensive software systems	850,000 sequences
	P50	Motion analysis	3×1000 sequences

TABLE A7 | Comparison of the nine most promising approaches (three RQ facets satisfied; evolution missing).

ID	Approach type	Input needed (no real-life raw data)	Domain/purpose	Output/generated data	Real-life like	Evolution	Tool support	Limitations	QA (R1/R2)
P01	Property-Based Testing with External Test-Case Generators https://doi.org/10.1109/ICSTW.2017.62								
	Rule-based from WSDL	WSDL specifications, properties	Test case generation for service interfaces	Test cases guided by interface constraints	Yes	No	MoMuT; FsCheck[a]	Higher compute; replicability description limited	3/4; 0/4
P03	Grammar-Based Testing Using Realistic Domains in PHP https://doi.org/10.1109/ICST.2012.136								
	Rule−/grammar-based	Grammars, tokens, contracts	Unit/API testing (textual inputs)	Realistic strings (e.g., emails) per grammar	Yes	No	Praspel[b]	Limitations not explicitly discussed	3/4; 1/4
P16	A SMART Approach to Quality Assessment of Site-Based Spatio-Temporal Data https://doi.org/10.1145/2996913.2996932								
	Successive Random Addition (SRA)	Randomised parameters (no real data)	Spatio-temporal data quality assessment	Weather-like synthetic phenomena	Yes	No	Not described	Not reported; generalizability and controls limited	2/4; 2/4
P28	Practical Constraint Solving for Generating System Test Data https://doi.org/10.1145/3381032								
	UML/OCL constraint solving (rule-based)	UML models + OCL constraints	System-level testing (various domains)	Large instance models (thousands of objects)	Yes	No	PLEDGE[c]	Constraint solver may be non-exhaustive	4/4; 4/4
P53	G-Tric: generating three-way synthetic datasets with triclustering solutions https://doi.org/10.1186/s12859-020-03925-4								
	Parametric generation with planted triclusters	User-defined dataset/ solution properties	Clustering/data mining evaluation	3-way datasets (obs × features × contexts)	Yes	No	G-Tric[d]	Computational cost for large planted structures	2/4; 4/4
P55	Methodology for Building Synthetic Datasets with Virtual Humans https://doi.org/10.1109/ISSC49989.2020.9180188								
	Image rendering (virtual humans)	3D assets; rendering pipeline	Face/biometrics detection & recognition	Photo-realistic face imagery (virtual humans)	Yes	No	iClone; Blender[e]	Limitations not explicitly discussed	2/4; 4/4
P68	First principle models based dataset generation for MTR/MLC evaluation CEUR-WS								
	Simulation environment	First-principles models + parameters	ML evaluation (multi-target regression, multi-label classification)	Synthetic datasets with controlled change events	Yes	No	Not described	Limited inputs/outputs; short paper; non-peer-reviewed venue[f]	2/4; 1/4
P69	Automatically extracting mock object behaviour from Design by Contract https://doi.org/10.1145/1808266.1808273								
	Rule-based from Design-by-Contract + mocking	DbC specs (no real data)	Unit testing/mocking	Valid test inputs & mock behaviours	Yes	No	Mocking (not detailed)	Performance and generated data quality concerns	2/4; 4/4
P75	A Rule-Based Software Test Data Generator https://doi.org/10.1109/69.75894								
	Rule-based generator	User-defined rules	Testing of large software systems	Rule-driven test cases	Yes	No	Not described	Not reported; early tool-era constraints	3/4; 3/4

[a]Tools available, but replicability not fully described in the paper.
[b]Tool referenced; appeared inactive at the time of the review.
[c]Prototype available on GitHub; noted as not active at the time of review.
[d]Tool accessible at the time of the review.
[e]Mix of commercial and open-source tools in rendering pipeline.
[f]Only paper in the set not published in a peer-reviewed venue.