BenchCouncil Transactions on Benchmarks, Standards and Evaluations

TBench
Volume 4, Issue 4
2024

Original Articles

ISSN: 2772-4859

BenchCouncil Transactions on Benchmarks, Standards and Evaluations (TBench) is an open-access multi-disciplinary journal dedicated to benchmarks, standards, evaluations, optimizations, and data sets. This journal is a peer-reviewed, subsidized open access journal where The International Open Benchmark Council pays the OA fee. Authors do not have to pay any open access publication fee. However, at least one of the authors must register BenchCouncil International Symposium on Benchmarking, Measuring and Optimizing (Bench) (https://www.benchcouncil.org/bench/) and present their work. It seeks a fast-track publication with an average turnaround time of one month.

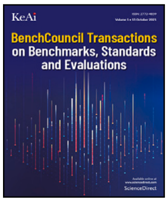
Contents

BenchCouncil Transactions on Benchmarks, Standards and Evaluations 4 (2024) 100190

Contents lists available at ScienceDirect

BenchCouncil Transactions on Benchmarks, Standards and Evaluations

journal homepage: www.keaipublishing.com/en/journals/benchcouncil-transactions-on-benchmarks-standards-and-evaluations/



Full length article

Open Source Evaluatology: An evaluation framework and methodology for open source ecosystems based on evaluatology

Fanyu Han [a], Shengyu Zhao [b], Wei Wang [a,*], Aoying Zhou [a], Weining Qian [a], Xuan Zhou [a], Jiaheng Peng [a], Lan You [c], Yang Chen [d], Xiaoya Xia [e], Yenan Tang [f], Liyun Yang [g], Chunqi Tian [b]

[a] *School of Data Science and Engineering, East China Normal University, Shanghai 200062, China*
[b] *School of Electronic and Information Engineering, Tongji University, Shanghai 200092, China*
[c] *School of Computer Science and Information Engineering, Hubei University, Hubei Wuhan 430062, China*
[d] *School of Computer Science, Fudan University, Shanghai 200438, China*
[e] *Ant Group, Zhejiang Hangzhou 310000, China*
[f] *Alibaba Group, Zhejiang Hangzhou 310099, China*
[g] *China Electronics Standardization Institute, Beijing 100007, China*

ARTICLE INFO

Keywords:
Open source evaluatology
Evaluatology
Open source ecosystem
Evaluation framework
Evaluation methodology

ABSTRACT

The open-source ecosystem, as an important component of the modern software industry, has increasingly attracted attention from both academia and industry regarding its evaluation. However, current open-source evaluation methods face several issues, such as inconsistent evaluation standards, lack of theoretical support in the evaluation process, and poor comparability of evaluation results. Guided by the foundational theories of evaluatology, this paper proposes a new interdisciplinary research field, Open Source Evaluatology, and constructs an evaluation theoretical framework and methodology for open-source ecosystems. The main contributions of this paper include: (1) Based on the five axioms of evaluation theory, a theoretical system for Open Source Evaluatology is developed, and the basic concepts, evaluation dimensions, and evaluation standards for the open-source ecosystem are proposed; (2) An evaluation conditions (EC) framework is designed, encompassing five levels: problem definition, task instances, algorithm mechanisms, implementation examples, and supporting systems. A combined evaluation model (EM) based on statistical metrics and network metrics is also introduced; (3) Experimental validation using the GitHub dataset shows that the proposed evaluation framework effectively assesses various features of open-source projects, developers, and communities, and has been verified in multiple practical application scenarios. The research demonstrates that Open Source Evaluatology provides a standardized theoretical guide and methodological support for open-source ecosystem evaluation, which can be widely applied in various scenarios, such as open-source project selection, developer evaluation, and community management, and plays a significant role in promoting the healthy and sustainable development of open-source ecosystems.

1. Introduction

In recent years, the open-source ecosystem has become a critical pillar of modern software development. The widespread use of open-source projects has not only driven technological innovation but also fostered global collaboration and knowledge sharing [1]. According to GitHub's "Octoverse 2024" report, more than 100 million developers worldwide are participating in open-source projects, with over 330 million hosted open-source projects.[1] The rapid development of the open-source ecosystem provides significant support for technological advancement and industrial transformation, but it also brings challenges in management and evaluation [2].

In the open-source ecosystem, evaluation plays a crucial role. On one hand, the quality and long-term value of open-source projects need to be scientifically assessed to enable developers [3], organizations [4], and investors to make informed decisions. On the other hand, the health of the community and the level of developer contributions also need to be measured to ensure the sustainable development of the open-source ecosystem [5]. For instance, companies need to evaluate the activity and stability of open-source projects when selecting technologies; open-source foundations need to assess the health of their communities [6]; and individual developers wish to gain recognition for their contributions through a scientific evaluation system [7].

[*] Corresponding author.
E-mail address: wwang@dase.ecnu.edu.cn (W. Wang).

[1] https://octoverse.github.com

https://doi.org/10.1016/j.tbench.2025.100190
Received 4 January 2025; Received in revised form 29 January 2025; Accepted 13 February 2025
Available online 23 February 2025

However, existing evaluation methods for the open-source ecosystem have several shortcomings. Most current research focuses on a single dimension, such as code quality [8], developer activity [9], or community governance [10], and lacks a systematic theoretical framework and a multidimensional comprehensive evaluation mechanism. Moreover, the definition of evaluation standards often relies on experience or specific contexts, lacking unified theoretical guidance, which results in poor comparability of evaluation results. These issues suggest that current evaluation methods are insufficient to meet the increasingly complex needs of the open-source ecosystem.

To address the aforementioned issues, this paper, guided by the fundamental theories of Evaluatology [11], introduces the novel interdisciplinary field of Open Source Evaluatology and conducts research focused on the evaluation needs of the open-source ecosystem. The specific objectives include:

- Establishing the theoretical system of Open Source Evaluatology: Based on the five basic axioms and the conditions framework of Evaluatology, this paper constructs a theoretical system for Open Source Evaluatology tailored to the open-source ecosystem, clarifying the basic concepts, dimensions, and standards of evaluation.
- Constructing the evaluation framework and methodology: The paper designs evaluation conditions (Evaluation Conditions, EC) and evaluation models (Evaluation Models, EM) that are adapted to the characteristics of the open-source ecosystem, and proposes a practical and actionable methodological system.
- Verification and application: Using the GitHub open-source dataset, the paper validates the effectiveness of the proposed framework and applies it to evaluate open-source projects, developers, and communities.

The structure of the remainder of this paper is as follows: Section 2 introduces the basic theories of Evaluatology and analyzes the characteristics of the open-source ecosystem, thereby proposing the theoretical framework of Open Source Evaluatology. Section 3 provides a detailed description of the evaluation framework and methodology, including the construction of evaluation conditions, the design of evaluation models, and the establishment of evaluation standards. Section 4 presents empirical research on the proposed framework through experimental design. Section 5 presents application verification, analyzing its effectiveness in evaluating open-source projects, developers, and communities. Section 6 summarizes the main research findings, discusses the innovations and limitations of this paper, and outlines future research directions.

2. The theoretical of open source evaluatology

2.1. The theoretical of evaluatology

Evaluation is the process of making value judgments about things, behaviors, or systems under specific conditions. Its core lies in systematically analyzing and comprehensively judging the strengths, weaknesses, gains, losses, or adaptability of the evaluation object through a set of scientific theories, methods, and standards. The essence of evaluation can be summarized in three aspects: (1) goal orientation, meaning that any evaluation activity must be conducted around a clear objective; (2) condition dependency, meaning that the evaluation results depend on the set conditions and environment; (3) standard normativity, meaning that the evaluation conclusion relies on an objective and consistent set of evaluation standards.

Evaluatology, as a theoretical discipline, is dedicated to revealing the fundamental laws of evaluation and constructing a system of evaluation theory, methods, and applications. In the open-source ecosystem, the core of evaluation is to measure the quality, activity, and sustainability of open-source projects, developers, and communities using both quantitative and qualitative methods, in order to support scientific decision-making and resource allocation.

The theoretical system of Evaluatology is based on five fundamental axioms, which provide a unified theoretical foundation for any evaluation activity.

- Existence Axiom: Any evaluation object, under specific conditions, is evaluable, meaning all objects can be value-judged according to certain rules.
- Condition Axiom: The results of an evaluation depend on defined conditions and environment, meaning evaluation outcomes may vary under different conditions.
- Object Axiom: Evaluation must target a clear object, and the properties and characteristics of the object are the core basis for the evaluation activity.
- Standard Axiom: Evaluation requires an objective and standardized set of criteria, and all evaluation results must be derived from this standard.
- Purpose Axiom: Every evaluation activity must focus on a specific goal, and achieving the evaluation goal is the ultimate direction.

In Open Source Evaluatology, these axioms are used to guide the construction of evaluation models and the design of evaluation criteria. For example, the Condition Axiom requires us to define the goal when evaluating an open-source project (such as selecting high-quality projects or predicting the project's lifecycle), while the Standard Axiom ensures that the evaluation results are consistent and comparable.

Evaluation Conditions (EC) are the core components of an evaluation activity. They define the relationship between the evaluation object and its environment and provide the foundational framework for designing the evaluation model. Evaluation conditions are typically divided into five components:

- Problem Definition Component: Defines the goals and scope of the evaluation activity. For example, in the open-source domain, the goal might be to assess the quality of a project or the health of a community.
- Task Instance Component: Breaks down the evaluation goal into specific tasks, such as evaluating code quality or developer activity.
- Algorithm Mechanism Component: Designs the algorithms or mechanisms used to accomplish the tasks, such as network analysis-based methods for evaluating community structure.
- Implementation Instance Component: Applies the algorithmic mechanisms to specific datasets or environments, such as using Git data for experimental validation.
- Support System Component: Provides the technical and environmental support necessary for the evaluation, such as data collection systems and computational platforms.

In Open Source Evaluatology, the hierarchical structure of evaluation conditions offers theoretical guidance for constructing a systematic evaluation framework.

2.2. Analysis of open source ecosystem characteristics

The open-source ecosystem is composed of developers, projects, organizations, and communities from around the world. Its collaboration models, data characteristics, and evaluation challenges provide researchers with rich research topics, while also presenting unique technical demands.

The collaboration model in the open-source ecosystem has distinct distributed characteristics. Developers collaborate across regions through online platforms such as GitHub, breaking the geographical and organizational boundaries of traditional software development [12]. This distributed collaboration model allows developers to participate in project development asynchronously. Whether through submitting code, fixing issues, or engaging in discussions, it effectively promotes the rapid iteration and innovation of open-source projects [13]. The open-source ecosystem exhibits significant diversity and dynamism [14]. Different open-source projects and communities vary widely in terms of scale, technology stack, and governance models. For example, some open-source communities rely on centralized decision-making by core maintainers, while others tend toward distributed consensus governance. Additionally, the projects and communities within the ecosystem change dynamically over time. For instance, the development activity of a project may significantly fluctuate due to technical trends or external resource support [15]. The open-source ecosystem is characterized by high transparency and openness [2]. Collaboration records, development processes, and codebases are typically public. This openness provides researchers with abundant data sources, making it possible to evaluate open-source projects and communities.

2.3. Theoretical framework of open source evaluatology

As shown in Fig. 1, Open Source Evaluatology, as the application of Evaluatology within the open-source ecosystem, demands a theoretical framework that is systematic, scientific, and adaptable to address the diverse and dynamic evaluation needs of the ecosystem. Theoretical of Open Source Evaluatology is directly derived from the foundational principles of Evaluatology, which is a theoretical discipline focused on establishing systematic evaluation frameworks. While Evaluatology provides a general framework for evaluation through five fundamental axioms (such as the Condition Axiom and Standard Axiom), Open Source Evaluatology applies these axioms to the unique context of open-source ecosystems. It adapts Evaluatology's core concepts—such as evaluation objects, conditions, and results—by addressing specific challenges in open-source environments, such as assessing project quality, developer activity, and community sustainability. Thus, Open Source Evaluatology is an extension of Evaluatology, applying its general evaluation principles to the dynamic and multifaceted nature of open-source projects. This section outlines the theoretical framework of Open Source Evaluatology from three perspectives: basic concepts, evaluation dimensions, and evaluation standards.

2.3.1. Definition of basic concepts

Based on the fundamental theories of Evaluatology, the core concepts in Open Source Evaluatology include evaluation objects, evaluation conditions, and evaluation results. Firstly, the evaluation object is a central element in Open Source Evaluatology, comprising three levels of evaluation units: open source projects, developers, and community organizations. Open source projects are the basic components of the open-source ecosystem, and their quality and activity directly impact the overall health of the ecosystem [16]. Developers, as core participants in projects, contribute in ways that affect the project's sustainable development [17]. Community organizations, as the organizational carriers of project collaboration, influence the collaboration efficiency and overall health of the open-source ecosystem, with their size, structure, and activity level being key factors.

Evaluation conditions are the prerequisites for conducting evaluation activities, including specific evaluation objectives, evaluation environments, and datasets. For example, different evaluation objectives (such as selecting high-quality projects or assessing developer influence) may require different evaluation standards and methods. Environmental variables (such as project technology stack or community size) can significantly influence evaluation results, while data sources (e.g., GitHub or Gitee) determine the feasibility and accuracy of evaluations.

Evaluation results represent the final output of an evaluation activity, typically manifested as value judgments about the evaluation object. For example, evaluations can identify which projects are high-quality, which developers have a higher community impact, and which communities are more vibrant and sustainable. These results provide references for developers and enterprises in technology selection and resource allocation, and offer scientific evidence for open-source governance and policy-making.

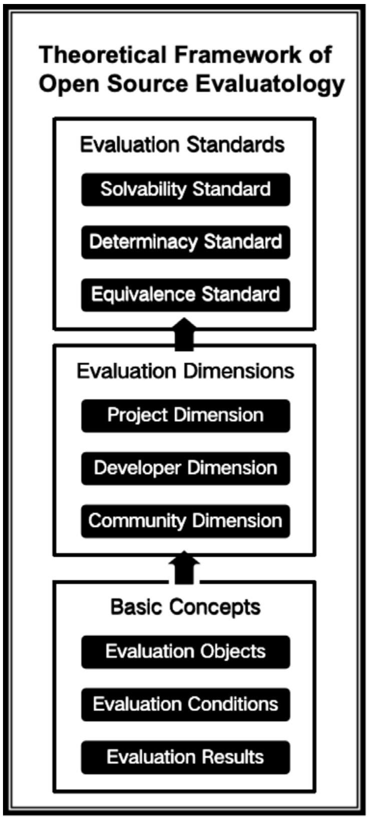


Fig. 1. Theoretical Framework of Open Source Evaluatology.

2.3.2. Design of evaluation dimensions

The design of evaluation dimensions in Open Source Evaluatology needs to reflect the characteristics and value of the evaluation object from different perspectives. Based on the structural characteristics of the open-source ecosystem, this study divides the evaluation dimensions into project, developer, and community dimensions.

Project Dimension: The project dimension mainly focuses on the quality, activity, and sustainability of open-source projects. Specific indicators include code quality, issue response time, and version release frequency.

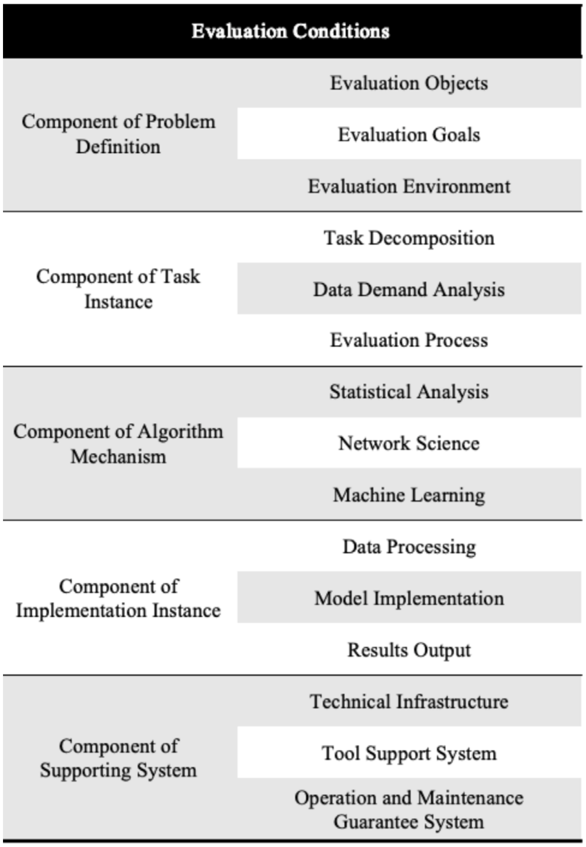
Developer Dimension: The developer dimension focuses on the quality of developers' contributions and their community influence. Specific indicators include activity level, contribution quality, and influence.

Community Dimension: The community dimension concerns the collaborative ecology and governance efficiency of open-source projects. Specific indicators include community size, structural stability, and collaboration efficiency.

2.3.3. Evaluation standards system

To ensure the scientificity and operability of open-source evaluations, Open Source Evaluatology needs to establish a unified evaluation standards system. According to the theoretical requirements of Evaluatology, this study proposes the following three types of evaluation standards:

Solvability Standard: The solvability standard requires that the evaluation task can be solved using existing methods.



Evaluation Conditions	
Component of Problem Definition	Evaluation Objects
	Evaluation Goals
	Evaluation Environment
Component of Task Instance	Task Decomposition
	Data Demand Analysis
	Evaluation Process
Component of Algorithm Mechanism	Statistical Analysis
	Network Science
	Machine Learning
Component of Implementation Instance	Data Processing
	Model Implementation
	Results Output
Component of Supporting System	Technical Infrastructure
	Tool Support System
	Operation and Maintenance Guarantee System

Fig. 2. Evaluation Conditions.

Determinacy Standard: The determinacy standard requires that evaluation results be consistent and reproducible.

Equivalence Standard: The equivalence standard requires that different evaluation methods yield similar results when handling the same task.

3. Framework and methods of open source evaluatology

3.1. Construction of evaluation conditions

Evaluation Conditions (EC) form the foundational framework of the open-source evaluation system, and the process of constructing these conditions directly impacts the scientific rigor and effectiveness of the evaluation activities. Based on the fundamental theories of Evaluatology, this study proposes a five-components evaluation conditions framework (as shown in Fig. 2), which systematically builds the theoretical foundation and practical guidelines for open-source evaluation, from problem definition to supporting systems.

3.1.1. Component of problem definition

The problem definition component is the top level design of the evaluation conditions framework, and its core task is to transform open-source evaluation requirements into formalized evaluation problems. In the open-source ecosystem, defining evaluation problems requires considering the diversity of evaluation objects, the complexity of evaluation goals, and the dynamic nature of the evaluation environment [18]. First, the definition of evaluation objects needs to clarify the evaluation units (such as projects, developers, or communities) and their attribute characteristics. For example, when evaluating an open-source project, multiple dimensions, such as the project's technical features, development stage, and application scenarios, need to be considered. Secondly, the setting of evaluation goals should be based on practical needs and be both actionable and measurable. For instance, project quality evaluation can be broken down into specific indicators like code quality, documentation completeness, and maintenance activity. Finally, the analysis of the evaluation environment needs to consider practical conditions such as data availability, computational resource constraints, and time requirements.

In the problem definition component, this study proposes a "Goal-Scenario-Indicator" three-dimensional analysis framework. In the goal dimension, a hierarchical structure for evaluation goals is established to decompose abstract evaluation requirements into specific evaluation tasks. In the scenario dimension, the feasibility of the evaluation plan is ensured by analyzing the application environment and constraints. In the indicator dimension, a scientific indicator system is built to provide a quantitative foundation for the evaluation activities. This systematic problem definition method not only improves the focus of evaluation activities but also provides clear guidance for the subsequent task implementation.

3.1.2. Component of task instance

The task instance component is responsible for transforming abstract evaluation problems into concrete execution tasks, acting as the key link between theoretical design and practical implementation. In open-source evaluation, the design of task instances needs to consider the complexity and systematism of evaluation activities, ensuring their effective execution through scientific task decomposition and process design.

Task decomposition is the core work of the task instance layer. Based on systems engineering principles, this study proposes a "hierarchical and graded" task decomposition method. On the horizontal dimension, evaluation tasks are divided into sub-tasks such as data collection, feature extraction, model calculation, and result presentation. On the vertical dimension, a multi-level task execution system is established based on task complexity and dependencies. This structured decomposition effectively reduces the complexity of evaluation activities and improves the efficiency and controllability of task execution.

Data demand analysis is an important part of task design. The data generated by the open-source ecosystem is characterized by multiple sources, heterogeneity, and dynamic changes. This requires clear data acquisition strategies, quality control standards, and processing workflows during the task design phase. This study designs a data demand analysis framework that includes key elements such as data source identification, quality assessment, acquisition plans, and preprocessing strategies, providing a systematic guide for subsequent data processing tasks.

The design of the evaluation process needs to consider task execution sequence, parallelism, and feedback mechanisms. Based on workflow theory, this study constructs a task execution framework that adapts to the characteristics of open-source evaluation, supporting parallel processing of tasks, intermediate result caching optimization, and handling of exceptions to ensure that the evaluation activities can proceed efficiently and stably.

3.1.3. Component of algorithm mechanism

The algorithm mechanism component is the technical core of the evaluation conditions framework. Its main task is to design and implement various algorithms and computational models required for evaluation. In open-source evaluation, the design of algorithm mechanisms needs to consider both the scientific nature of the theory and the feasibility of practical implementation, ensuring the accuracy and reliability of evaluation results through appropriate technical choices and optimization designs.

In terms of basic algorithm design, this study integrates methods from multiple fields, including statistical analysis, network science, and machine learning. Statistical analysis methods are mainly used for descriptive data analysis and preliminary modeling, including distribution characteristic analysis, correlation analysis, and time-series analysis. Network analysis algorithms are used to handle complex relational networks in the open-source ecosystem, including community detection, centrality calculation, and influence evaluation. Machine learning models are employed to address complex evaluation tasks such as project quality prediction and developer behavior analysis.

3.1.4. *Component of implementation instance*

The implementation instance component is responsible for transforming theoretical designs into executable technical solutions, representing the practical aspect of the evaluation conditions framework. This design must consider the constraints and needs of the actual application environment to ensure the usability and maintainability of the evaluation methods.

In data processing, this study establishes a complete data lifecycle management system. First, a distributed crawler system is designed to efficiently collect data from multiple open-source platforms (e.g., GitHub, GitLab). Next, automated data quality control tools are developed to identify and process anomalous data, duplicate data, and missing values. Finally, an extensible feature extraction framework is constructed to support the rapid implementation of new feature computation requirements.

Model implementation is the core task of the implementation layer. This study adopts a modular and object-oriented design concept to build a unified model implementation framework, providing standardized interfaces for model integration and expansion, and includes a complete testing and verification mechanism to ensure correctness. The results output phase focuses on the interpretability and usability of evaluation results. A multi-level result presentation system is designed, including numerical results, statistical charts, and interactive visualizations.

3.1.5. *Component of supporting system*

The supporting system component provides the infrastructure and technical support for evaluation activities, ensuring the stable operation of the evaluation system. The design of this component must comprehensively consider the system's usability, scalability, and maintainability.

Technical infrastructure serves as the physical foundation for the supporting system, while the tool support system enhances the efficiency of evaluation activities. The operation and maintenance guarantee system ensures the stable functioning of the system.

3.2. *Evaluation model*

The evaluation model (EM) is a core component of the open-source evaluation system, and its design directly determines the scientific accuracy and reliability of the evaluation results. Based on the characteristics of the open-source ecosystem, this study constructs a multidimensional evaluation model system, which includes two main parts: the statistical measurement model [19] and the network measurement model [20].

3.2.1. *Statistical measurement model*

The statistical measurement model primarily focuses on the static characteristics of the evaluation objects. In the evaluation of open-source projects, these indicators include basic metrics such as code volume, commit frequency, and issue response time; in developer evaluation, it includes activity indicators such as the number of contributions, amount of code changes, and number of comments; in community evaluation, it includes structural indicators such as member size, geographical distribution, and activity level. The above evaluation metrics are defined by the CHAOSS community,[2] which is an open-source community dedicated to creating metrics and models for evaluating open-source software projects and their ecosystems. These basic metrics are standardized and normalized using scientific statistical methods to ensure comparability between evaluation objects of different scales and types.

In addition to the aforementioned indicators, this paper proposes a scalable multidimensional activity quantification metric, calculated by a weighted sum of developer behaviors based on GitHub event counts. The Activity metric has been integrated into OpenDigger.[3] For a dataset D within a certain time period (from t_{start} to t_{end}), a set of behaviors B_k can influence the activity of GitHub developer C. K represents the total number of types of behaviors that affect activity, and k refers to a specific type of behavior. The AGGREGATE function groups and aggregates C's behaviors in dataset D. Different weight values W_{B_k} are assigned based on the importance of each behavior. The activity metric of C is the average over the specified time period, calculated by dividing the weighted sum of behaviors by the number of days.

$$Activity_C = \frac{\sum_{k=1}^{K} W_{B_k} AGGREGATE(B_k, D)}{t_{end} - t_{start}} \tag{1}$$

To determine the weights, we applied the Analytic Hierarchy Process (AHP), which is commonly used in operations research [21]. Weight determination is subjective to experts' experience, especially tenured developers in open source communities. Thus, we presented the AHP evaluation matrix (as shown in Table 1) to the seven maintainers of participated projects, instructed them on the 5-point scale AHP evaluation, and then calculated the final value by the geometric mean. Using AHP analysis, we obtained the weights for each collaboration behavior. And correspondingly, the activity metric of any repository R is the sum of the square roots of all the developers' activity metric value who collaborated during the time period. The calculation of the square root is intended to mitigate the impact of excessively high activity levels from individual developers, such as automated accounts, thereby indirectly incorporating the number of active developers within the community into this metric. I represents the total number of developers who have contributed to the repository.

$$Activity_R = \sum_{i=1}^{I} \sqrt{Activity_C} \tag{2}$$

3.2.2. *Network measurement model*

For event data on platforms like GitHub, where each record corresponds to a specific action by a developer at a certain time in a repository, a time-sequenced, heterogeneous collaboration network can be constructed using developers and repositories as nodes, and the developers' actions as edges. Building on the Activity metric, this paper has introduced a new network metric called OpenRank.

The OpenRank metric constructs a global collaboration network(Fig. 3) with developers and repositories as nodes, and the Activity metric of developers on repositories as edges. It uses an algorithm similar to PageRank to measure the centrality of all nodes. Unlike the classic PageRank algorithm, the OpenRank algorithm allows nodes to partially depend on their initial values during centrality calculation. Although this dependency means that OpenRank's computation is no longer a Markov process, it also broadens its application scope. In calculating OpenRank, OpenDigger constructs the global collaboration network on a monthly basis, and the calculation allows all nodes to use the results from the previous month as their initial values.

$$\boldsymbol{c} = (\boldsymbol{E} - \boldsymbol{A}\boldsymbol{S})^{-1}(\boldsymbol{E} - \boldsymbol{A})\boldsymbol{c}^{(0)} \tag{3}$$

The computation result of OpenRank is represented as (3), where $\boldsymbol{E}$ is the identity matrix, $\boldsymbol{S}$ is the linearly normalized connection matrix, $\boldsymbol{A}$ is the matrix representing the degree of dependency on initial values,

[2] https://chaoss.community/kb-metrics-and-metrics-models/

[3] https://github.com/X-lab2017/open-digger

Table 1
AHP evaluation matrix and results.

Behavior	Issue/PR comment	PR review	Close issue	Close PR	Open issue	Open PR	Eigenvector	Weight(%)
Issue comment	1	0.5	0.5	0.333	0.25	0.2	0.401	5.252
PR review	2	1	0.5	0.5	0.333	0.2	0.567	7.427
Close issue	2	2	1	0.5	0.333	0.25	0.742	9.712
Close PR	3	2	2	1	0.5	0.333	1.222	14.695
Open issue	4	3	3	2	1	0.333	1.698	22.235
Open PR	5	5	4	3	3	1	3.107	40.679

and $\boldsymbol{c}^{(0)}$ is the vector of initial values for all nodes. The vector $\boldsymbol{c}$ represents the node values after multiple iterations, when the algorithm has converged. For a proof of convergence of the OpenRank algorithm that supports initial values, one can refer to the OpenRank Leaderboard paper [20].

The core idea of the OpenRank algorithm is the collaborative evaluation between repository nodes and developer nodes. A repository with a high OpenRank indicates that it is active with many developers who themselves have high OpenRank values; similarly, a high OpenRank value for a developer indicates that the developer consistently contributes to repositories with high OpenRank values. The introduction of initial values means that OpenRank results have temporal continuity, allowing long-term contributions to be observed rather than relying solely on activity within the current month.

3.3. Evaluation standards

Evaluation standards are essential guidelines for regulating the evaluation process, unifying evaluation dimensions, and interpreting the results in the open-source evaluation system. Scientific and reasonable evaluation standards ensure the transparency, objectivity, and comparability of the evaluation activities. In open-source ecosystems, due to the diversity of evaluation objects and the complexity of application scenarios, the establishment of evaluation standards must comprehensively consider theoretical foundations, industry practices, and practical needs. This study elaborates on the process of establishing evaluation standards from three aspects: framework construction, indicator weight allocation, and calibration mechanism design.

3.3.1. Framework construction

The construction of a standard framework is the primary task in establishing evaluation standards, with the core goal being to provide systematic guidance for complex open-source evaluations. This study proposes an evaluation standard framework based on hierarchical design, which includes three main levels: objective layer, indicator layer, and measurement layer.

- Objective Layer: Clarifies the macro objectives of the evaluation. Based on the characteristics of the open-source ecosystem, the evaluation goals can be divided into core directions such as quality management, capability assessment, and ecological health monitoring. For example, in open-source project evaluation, the core objective is to assess the project's technical maturity and sustainability. In developer evaluation, the focus is on analyzing the developer's contribution ability and collaboration ability.
- Indicator Layer: Breaks down the macro objectives from the objective layer into specific evaluation dimensions. The design of the indicator layer should comprehensively cover the core characteristics of the evaluation object, ensuring independence and complementarity among the dimensions. For example, the indicator layer for project evaluation could include dimensions such as code quality, activity, community involvement, and technical innovation. The indicator layer for developer evaluation could include dimensions such as development activity, code influence, and collaboration ability.
- Measurement Layer: For each evaluation dimension in the indicator layer, defines specific measurement standards and calculation methods. The measurement layer needs to integrate statistical analysis, network analysis, and other technical methods to convert abstract indicators into actionable measurements. For example, project activity can be measured by submission frequency, issue closure time, etc., while developer collaboration ability can be calculated using network centrality or team contribution ratio.

The hierarchical design of the standard framework ensures the logical consistency and systematization of the evaluation process, while also providing clear guidance for subsequent indicator weight allocation and calculations.

3.3.2. Indicator weight allocation

In multi-indicator comprehensive evaluation, different indicators may have varying impacts on the final evaluation result, so reasonable weight distribution is required. This study combines expert evaluation, analytic hierarchy process (AHP), and data-driven methods to propose a mixed weight allocation mechanism:

- Expert Evaluation: Experts in the open-source field score the importance of each indicator from an experience and practical perspective. Expert evaluation considers domain knowledge and industry needs, making it a crucial reference for standardized weight allocation.
- Analytic Hierarchy Process (AHP) [22]: Constructs pairwise comparison matrices for indicators to calculate their relative weights. The AHP method quantifies subjective judgments into mathematical forms, making it suitable for weight allocation in hierarchical indicator systems.
- Data-Driven Method: Based on historical evaluation data, correlation analysis, multivariate regression, and other statistical methods are used to determine the actual contribution of each indicator to the evaluation result. This data-driven method reflects the importance of indicators in practical applications, compensating for the subjectivity in expert evaluations.

By combining these three methods, this study designs a weight optimization framework that organically integrates expert knowledge with data-driven results, generating a comprehensive weight allocation scheme suitable for different scenarios. Additionally, to address the dynamic changes of evaluation objects, the weight optimization framework supports periodic updates to the weights, ensuring the timeliness and adaptability of the evaluation standards.

3.3.3. Calibration mechanism design

The calibration mechanism of evaluation standards is a key link in ensuring the accuracy and fairness of the evaluation results. Considering the complexity and diversity of the open-source ecosystem, this study designs a multi-dimensional calibration mechanism, including data calibration, model calibration, and result calibration:

- Data Calibration: During data collection and processing, standardization methods are used to unify data from different sources and formats, ensuring comparability between different data sources.

Table 2
Spearman's correlation of metrics.

	Activity	Activity^2	OpenRank	Participants	Open issue	Close issue	Open PR	Close PR
Activity	1	0.844	0.83	0.902	0.694	0.466	0.51	0.658
Activity^2	0.844	1	0.788	0.587	0.603	0.669	0.688	0.616
OpenRank	0.83	0.788	1	0.693	0.49	0.585	0.64	0.479

* All correlations in the table are significant, with $p < 0.01$.

For example, for activity data from different open-source platforms, normalization processing is applied to eliminate scale differences, and time alignment methods are used to resolve time window inconsistencies.

- Model Calibration: During model construction and calculation, optimization and debugging of model parameters are carried out to ensure the robustness and applicability of the evaluation model. For example, in different evaluation scenarios, algorithm parameters are dynamically adjusted to adapt to specific object characteristics and scenario requirements.
- Result Calibration: During the evaluation result output stage, comparison analysis and confidence interval calculation are introduced to verify the reliability of the evaluation results. For example, by comparing with historical evaluation results, anomalous fluctuations are identified, and by introducing uncertainty analysis, the credibility of the evaluation results is quantified.

Additionally, this study designs a benchmarking mechanism that compares the evaluation system's results with industry-recognized benchmark data, further improving the credibility of the evaluation standards.

4. Experimental validation

4.1. Experimental design

To conduct experimental research on the Activity and OpenRank evaluation models, this study utilized GitHub developer behavior data provided by OpenDigger, covering the period from January to June 2024. The data was analyzed monthly, calculating the Activity and OpenRank metrics for all repositories globally. Subsequently, based on the OpenRank metric, the top 1000 repositories worldwide with the highest OpenRank for each month, along with their corresponding statistical metrics, were selected for further discussion.

In the study of the Activity metric, an additional metric, Activity squared, was computed. This metric calculates the activity results of the repository without taking the square root of the developer's activity level, aiming to observe the impact of the square root operation on the Activity metric results.

4.2. Experimental analysis

From January to June 2024, the top 1000 repositories were extracted monthly using the OpenRank metric, with 1380 remaining after deduplication. Since these metrics were calculated monthly, each repository contributed one set of data points per month, resulting in a total of 8082 data records. Each data record includes 8 metrics for a specific month: Activity, Activity2, OpenRank, Participants, Open Issue, Close Issue, Open PR, and Close PR. The "Participants" metric refers to the number of developers who had at least one collaboration event during that month. Spearman's rank correlation is used to assess the monotonic relationship between two variables, making it suitable for non-linear or non-parametric data. A Spearman coefficient close to +1 indicates a strong positive monotonic relationship, while a coefficient close to −1 suggests a strong negative monotonic relationship, and a coefficient near 0 implies little to no monotonic correlation [23]. This study performed a Spearman correlation analysis on the metric data for all these records, with the analysis results shown in Table 2.

Table 3
Activity Metric Cases.

	Cityofaustin/atd-data-tech	Coinhall/yacar
Activity	737.87	194.35
Rank_a	#616	#4757
Activity^2	20 900.65	20 672.82
Rank_{a2}	#265	#273
Participants	39	13
Open issue	489	0
Open PR	0	1116

As demonstrated in the Table 2, the Activity metric and Activity^2 metric for all repositories show a significant positive correlation with all other related collaboration metrics. Since the Activity^2 metric does not involve taking the square root of the developers' Activity metric, it exhibits strong positive correlations (all higher than 0.6) with all collaboration metrics. In contrast, the Activity metric, due to the square root operation, generally shows lower correlations with collaboration metrics. However, the square root operation in the Activity metric is intended to mitigate the impact of excessively high levels of individual activity, thereby indirectly incorporating the factor of developer count. The Spearman's correlation analysis supports this rationale positively. The metric most highly correlated with the Activity metric is the number of community participants, reaching 0.902. Conversely, the number of participants is the metric with the lowest correlation in the Activity^2 metric, at only 0.587. This indicates that the square root operation indeed introduces the influence of the number of participants while ensuring a positive correlation with other collaboration metrics.

As shown in the Table 3, the repositories *cityofaustin/atd-data-tech* and *coinhall/yacar* are ranked 265th and 274th under the Activity^2 metric, respectively. Both repositories include accounts that exhibit automated behavior, with the former primarily focused on submitting issues and the latter on submitting PRs, as can also be seen from their statistical metrics. Correspondingly, under the Activity metric, these two repositories are ranked 616th and 4757th. The exceptionally high frequency of collaborative behavior brought by automated accounts is significantly mitigated by the square root operation, thus demonstrating the effectiveness of the Activity metric. Compared to the linear weighted sum approach, it can more effectively filter out open source repositories with larger collaboration scale.

As shown in the Table 2, the OpenRank metric, as a network metric, exhibits significant positive correlations with all other statistical metrics, though its correlations are generally lower than those of the Activity metric. However, since OpenRank utilizes collaborative relationships to construct its network, it extracts network information that cannot be gleaned from statistical metrics alone and considers the importance of developers when calculating the OpenRank of repositories, making it more effective in long-term collaborative network measurements.

This paper will illustrate the effectiveness of OpenRank through two repository cases. As shown in the Table 4, as one of the cases is from Alibaba Group, and data from September 2022 is used for illustration, a month that presents two contrasting cases. One is the *first-contributions/first-contributions*, a tutorial repository primarily designed to help developers new to the GitHub platform submit their first PR through detailed tutorials, thereby learning the pull-based workflow. This repository has thousands of developers participating in learning and PR submission each month, but these developers are typically

Table 4
OpenRank metric cases.

	Firstcontributions/first-contributions	Ice-lab/ice-next
Activity	7035.14	162.95
Rank_a	#6	#5472
OpenRank	481.63	27.99
Rank_o	#13	#2290
Participants	1631	11
Open issue	18	30
Open PR	1636	48

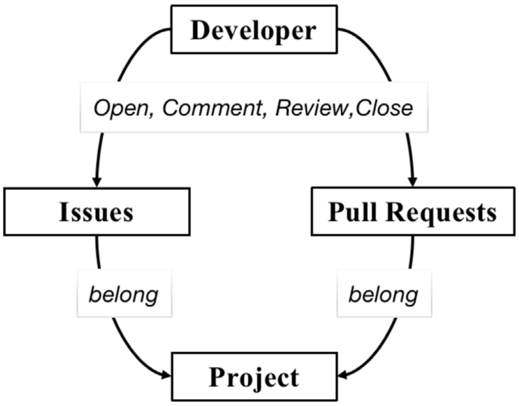


Fig. 3. Network Model.

newcomers to GitHub without prior collaboration or contribution experience on other repositories. Therefore, when using the OpenRank network metric, its ranking is noticeably lower than the Activity metric, dropping from 6th to 13th. The other case is the *ice-lab/ice-next* from Alibaba Group, a public but not externally promoted repository used by the ICE maintainer team for collaborative development of the next major version of the project. This repository started in early 2022 with consistently low levels of participation and low statistical metrics. However, since the participants are core maintainers of the ICE project, who had high OpenRank value due to their long-term contributions to the ICE repository by 2022, this repository, although a new one with few participants, saw a rapid increase in its OpenRank. Compared to its Activity rank of 5472, the OpenRank showed a significant improvement, ranking at 2290.

In these two cases, one involving numerous newcomers and the other involving a few veteran developers, the OpenRank's method of evaluating repository and developer collaboration provides corrections to the results that align with expectations, in contrast to the Activity metric.

5. Application cases

5.1. Project evaluation case

The evaluation model presented in this paper also serves as an open-source governance dashboard for enterprises, communities, foundations, and individual developers, providing open-source digital insights for OSPO (Open Source Program Office) practitioners and open-source project operators.

Fig. 4 shows the governance dashboard of the Alibaba/Nacos project, implemented using data services provided by OpenDigger and the DataV technology stack. This dashboard displays the activity and influence trends of the Nacos project, the number of participants, issue status, and PR activities. These insights help enterprises or organizations promptly understand the health, trends, and potential issues of their open-source projects and the overall community ecosystem, enabling more informed decision-making and optimization of operational strategies.

In addition to the dashboard shown in Fig. 4, enterprises can also implement multi-project competitive analysis dashboards to gain key insights into open-source projects of the same technology type, aiding in better technology selection. Using the evaluation model, data-driven insights and evaluation solutions can be applied across various levels, including developer-level, project-level, community organization-level, and foundation-level evaluations.

5.2. Developer evaluation case

Currently, many enterprises use evaluation models from open-source evaluation theory to assess the influence of developers involved in their open-source projects, thus creating incentives to encourage more newcomers to participate in open-source project development [19], as shown in Fig. 5.

Alibaba actively promotes the initiation and donation of several open-source projects, invests significant staff resources in operations and maintenance, and attracts external contributors to build a thriving open-source community. The company employs traditional open-source community management strategies, such as publishing technical articles, maintaining communication channels, and organizing events, to attract more developers to participate. However, these operational practices face challenges such as high costs and low conversion rates, making it difficult to intuitively assess a developer's community influence. Therefore, the industry requires a quantifiable approach to evaluate the contributions and value of developers, allowing for continuous motivation based on their actual contributions.

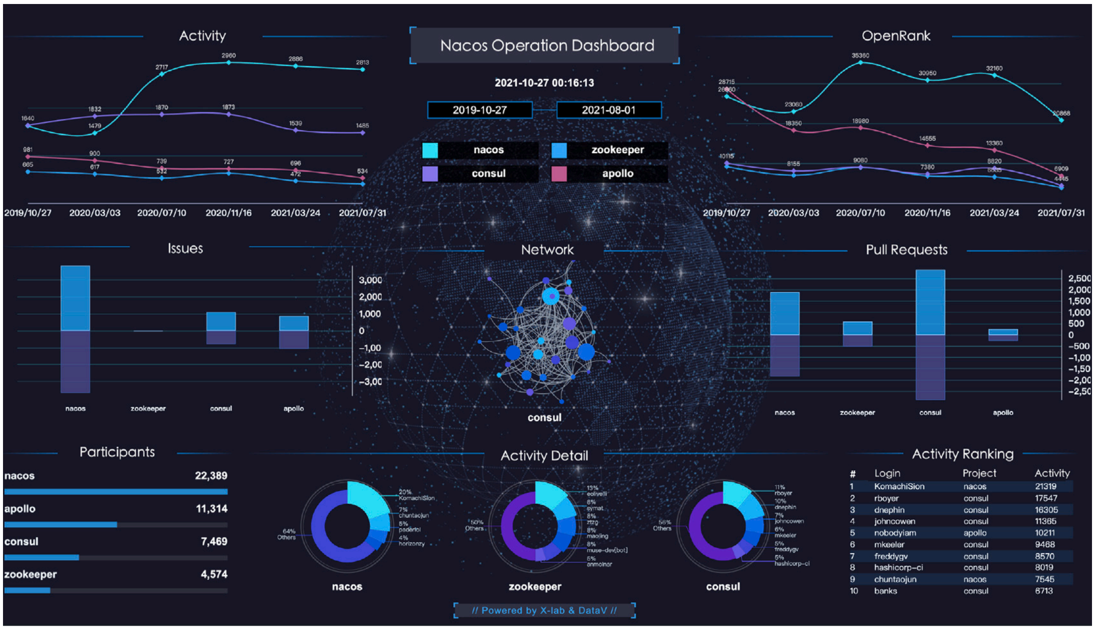
Alibaba's Open-Source Developer Contribution Incentive Leaderboard is updated monthly, offering rewards based on each developer's influence score. For example, when a developer's influence score reaches 50, they can exchange it for certain merchandise (e.g., T-shirts, keyboards, mice, etc.). A reasonable evaluation of open-source developers' individual influence can create incentives, further promoting the healthy development of the open-source ecosystem.

5.3. Community evaluation case

The Mulan Open Source Community focuses on monitoring its own health. The community applied the community evaluation method proposed in this study and identified issues such as decreased member participation and increased internal conflicts. Based on these findings, the community formulated targeted operational strategies. These series of standards have incorporated indicators, algorithm models, and information services from open-source evaluation theory as reference implementations, and have been gradually promoted for industry-wide adoption. As shown in Fig. 6, the Mulan Community developed a project incubation and governance dashboard based on evaluation theory methods, providing information services for many of its projects.

6. Conclusion

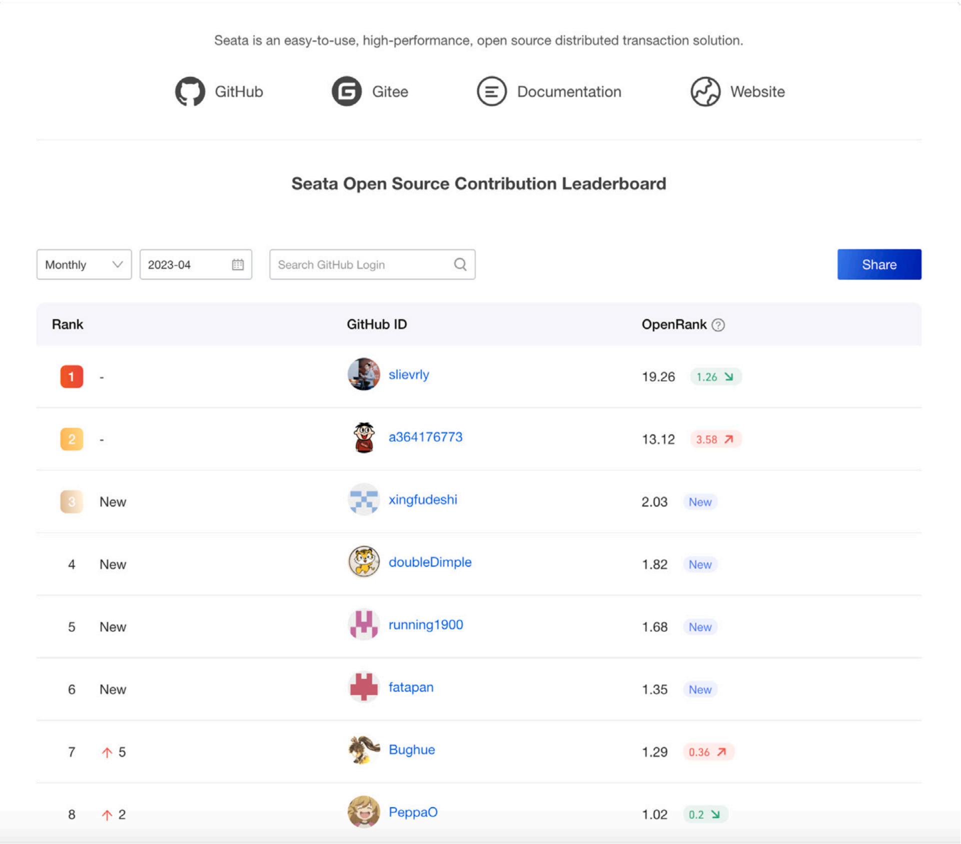
Based on the theory of evaluatology, this paper proposes a new interdisciplinary research field: Open Source Evaluatology. By constructing the theoretical framework of Open Source Evaluatology, designing the evaluation condition framework and evaluation models, and conducting experimental validation and application analysis, this study provides normative theoretical guidance and methodological support for the scientific evaluation of the open-source ecosystem. This research introduces the concept of Open Source Evaluatology for the first time, laying a solid theoretical foundation for this emerging interdisciplinary field; constructs a systematic evaluation condition framework and multi-level evaluation models; integrates techniques



Activity Ranking

#	Login	Project	Activity
1	KomachiSion	nacos	21319
2	rboyer	consul	17547
3	dnephin	consul	16305
4	johncowen	consul	11365
5	nobodyiam	apollo	10211
6	mkeeler	consul	9468
7	freddygv	consul	8570
8	hashicorp-ci	consul	8019
9	chuntaojun	nacos	7545
10	banks	consul	6713

Fig. 4. Nacos Operation Dashboard.



Seata is an easy-to-use, high-performance, open source distributed transaction solution.

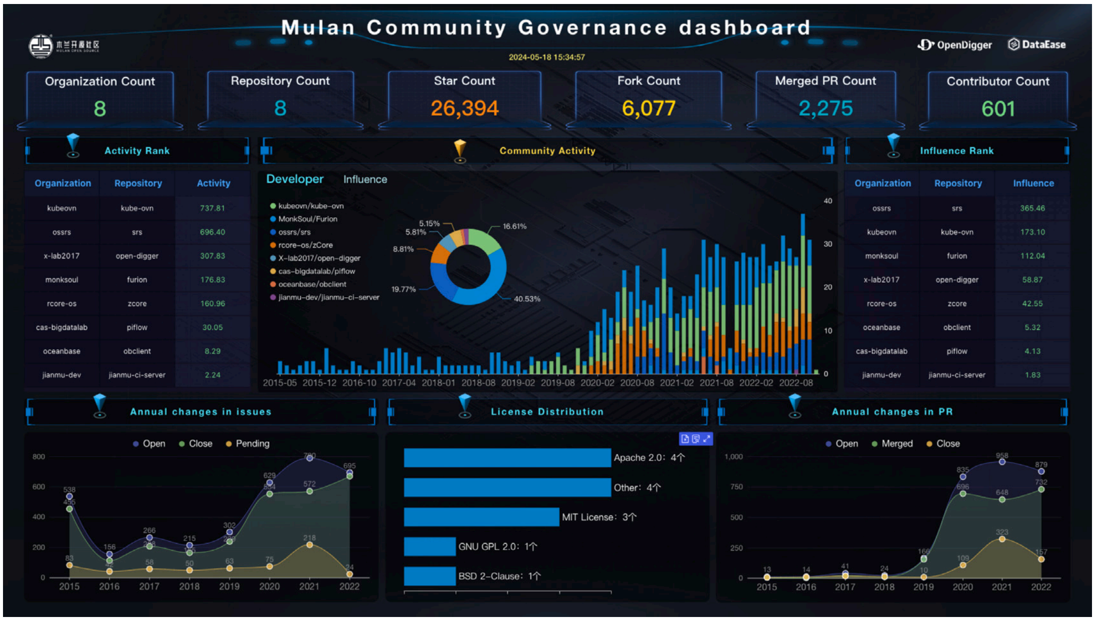
Seata Open Source Contribution Leaderboard

Rank		GitHub ID	OpenRank	
1	-	slievrly	19.26	1.26 ↘
2	-	a364176773	13.12	3.58 ↗
3	New	xingfudeshi	2.03	New
4	New	doubleDimple	1.82	New
5	New	running1900	1.68	New
6	New	fatapan	1.35	New
7	↑ 5	Bughue	1.29	0.36 ↗
8	↑ 2	PeppaO	1.02	0.2 ↘

Fig. 5. Alibaba Open Source Contribution Leaderboard.

such as statistical analysis and network analysis to improve the accuracy and interpretability of evaluation results; and demonstrates the effectiveness and practicality of the proposed methods through experimental validation and application cases.

Despite these achievements, this research still has some limitations that require further exploration. The adaptability of the evaluation models needs to be enhanced to cope with the dynamic changes in the open-source ecosystem. The interpretability and visualization of evaluation results need further optimization to improve user comprehension and experience. In the future, this research plans to apply Open Source Evaluatology to a wider range of scenarios, such as open-source project risk prediction, developer incentive mechanism design, and explore interdisciplinary integration with other related fields, in order



Mulan Community Governance dashboard

2024-05-18 15:34:57

Organization Count	Repository Count	Star Count	Fork Count	Merged PR Count	Contributor Count
8	8	26,394	6,077	2,275	601

Activity Rank

Organization	Repository	Activity
kubeovn	kube-ovn	737.81
ossrs	srs	696.40
x-lab2017	open-digger	307.83
monksoul	furion	176.83
rcore-os	zcore	160.96
cas-bigdatalab	piflow	30.05
oceanbase	obclient	8.29
jianmu-dev	jianmu-ci-server	2.24

Influence Rank

Organization	Repository	Influence
ossrs	srs	365.46
kubeovn	kube-ovn	173.10
monksoul	furion	112.04
x-lab2017	open-digger	58.87
rcore-os	zcore	42.55
oceanbase	obclient	5.32
cas-bigdatalab	piflow	4.13
jianmu-dev	jianmu-ci-server	1.83

Fig. 6. Mulan Community Governance Dashboard.

to provide stronger theoretical support and methodological guidance for the healthy development of the open-source ecosystem.

CRediT authorship contribution statement

Fanyu Han: Writing – original draft, Software, Data curation, Conceptualization. **Shengyu Zhao:** Software, Methodology, Data curation, Conceptualization. **Wei Wang:** Writing – review & editing, Supervision, Funding acquisition, Conceptualization. **Aoying Zhou:** Conceptualization, Supervision. **Weining Qian:** Conceptualization, Supervision. **Xuan Zhou:** Conceptualization, Supervision. **Jiaheng Peng:** Writing – review & editing, Visualization, Validation. **Lan You:** Writing – review & editing. **Yang Chen:** Methodology, Writing – review & editing. **Xiaoya Xia:** Writing – review & editing. **Yenan Tang:** Software. **Liyun Yang:** Supervision. **Chunqi Tian:** Writing – review & editing.

Declaration of competing interest

Xiaoya Xia is currently employed by Ant Group, Yenan Tang is currently employed by Alibaba Group and Liyun Yang is currently employed by China Electronics Standardization Institute. Other authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work is supported by the National Natural Science Foundation of China Grants 62137001, 62277017, and 61977026.

References

[1] J. West, S. Gallagher, Challenges of open innovation: the paradox of firm investment in open-source software, R & D Manage. 36 (3) (2006) 319–331.
[2] O. Franco-Bedoya, D. Ameller, D. Costal, X. Franch, Open source software ecosystems: A systematic mapping, Inf. Softw. Technol. 91 (2017) 160–185.
[3] W. Scacchi, Free/open source software development: Recent research results and methods, Adv. Comput. 69 (2007) 243–295.
[4] D. Woods, G. Guliani, Open Source for the Enterprise: Managing Risks, Reaping Rewards, "O'Reilly Media, Inc.", 2005.
[5] A. Mockus, R.T. Fielding, J.D. Herbsleb, Two case studies of open source software development: Apache and mozilla, ACM Trans. Softw. Eng. Methodol. (TOSEM) 11 (3) (2002) 309–346.
[6] K. Crowston, J. Howison, Assessing the health of open source communities, Computer 39 (5) (2006) 89–91.
[7] T. Devriendt, P. Borry, M. Shabani, Credit and recognition for contributions to data-sharing platforms among cohort holders and platform developers in Europe: interview study, J. Med. Internet Res. 24 (1) (2022) e25983.
[8] C.C. Silva, M. Galster, F. Gilson, Topic modeling in software engineering research, Empir. Softw. Eng. 26 (6) (2021) 120.
[9] D.M. German, The GNOME project: a case study of open source, global software development, Softw. Process: Improv. Pr. 8 (4) (2003) 201–215.
[10] J. Feller, Perspectives on Free and Open Source Software, MIT Press, 2005.
[11] J. Zhan, L. Wang, W. Gao, H. Li, C. Wang, Y. Huang, Y. Li, Z. Yang, G. Kang, C. Luo, et al., Evaluatology: The science and engineering of evaluation, BenchCouncil Trans. Benchmarks, Stand. Eval. 4 (1) (2024) 100162.
[12] E. Kalliamvakou, D. Damian, K. Blincoe, L. Singer, D.M. German, Open source-style collaborative development practices in commercial projects using GitHub, in: 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, vol. 1, IEEE, 2015, pp. 574–585.
[13] E. Raymond, The cathedral and the bazaar, Knowl. Technol. Policy 12 (3) (1999) 23–49.
[14] Y. Zhang, M. Zhou, K.-J. Stol, J. Wu, Z. Jin, How do companies collaborate in open source ecosystems? an empirical study of openstack, in: Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering, 2020, pp. 1196–1208.
[15] K. Crowston, K. Wei, J. Howison, A. Wiggins, Free/libre open-source software development: What we know and what we do not know, ACM Comput. Surv. 44 (2) (2008) 1–35.
[16] S. Jansen, Measuring the health of open source software ecosystems: Beyond the scope of project health, Inf. Softw. Technol. 56 (11) (2014) 1508–1519.
[17] Y. Fang, D. Neufeld, Understanding sustained participation in open source software projects, J. Manage. Inf. Syst. 25 (4) (2009) 9–50.
[18] D.A. Tamburri, F. Palomba, A. Serebrenik, A. Zaidman, Discovering community patterns in open-source: a systematic approach and its evaluation, Empir. Softw. Eng. 24 (2019) 1369–1417.
[19] X. Xia, Z. Weng, W. Wang, S. Zhao, Exploring activity and contributors on github: Who, what, when, and where, in: 2022 29th Asia-Pacific Software Engineering Conference, APSEC, IEEE, 2022, pp. 11–20.
[20] S. Zhao, X. Xia, B. Fitzgerald, X. Li, V. Lenarduzzi, D. Taibi, R. Wang, W. Wang, C. Tian, OpenRank leaderboard: Motivating open source collaborations through social network evaluation in Alibaba, in: Proceedings of the 46th International Conference on Software Engineering: Software Engineering in Practice, 2024, pp. 346–357.
[21] A. Darko, A.P.C. Chan, E.E. Ameyaw, E.K. Owusu, E. Pärn, D.J. Edwards, Review of application of Analytic Hierarchy Process (AHP) in construction, Int. J. Constr. Manag. 19 (5) (2019) 436–452.
[22] W. Ossadnik, O. Lange, AHP-based evaluation of AHP-software, European J. Oper. Res. 118 (3) (1999) 578–588.
[23] J.H. Zar, Significance testing of the Spearman rank correlation coefficient, J. Amer. Statist. Assoc. 67 (339) (1972) 578–580.

Contents lists available at ScienceDirect

BenchCouncil Transactions on Benchmarks, Standards and Evaluations

journal homepage: www.keaipublishing.com/en/journals/benchcouncil-transactions-on-benchmarks-standards-and-evaluations/

Full Length Article

COADBench: A benchmark for revealing the relationship between AI models and clinical outcomes

Jiyue Xie [a], Wenjing Liu [b],[*], Li Ma [b], Caiqin Yao [c], Qi Liang [a], Suqin Tang [a], Yunyou Huang [a]

[a] Key Lab of Education Blockchain and Intelligent Technology, Ministry of Education, Guangxi Normal University, No. 15 Yucai Road, Qixing District, Guilin, 541004, Guangxi, China

[b] Guilin Medical University, 20 Lequn Road, Xiufeng District, Guilin City, 541001, Guangxi, China

[c] The Sencond Nanning People's Hospital, No. 13, Dan Village Road, Jiangnan District, 530000, Guangxi, China

ARTICLE INFO

Keywords:
Alzheimer's disease
Benchmark
Clinical outcome
Artificial intelligence

ABSTRACT

Alzheimer's disease (AD), due to its irreversible nature and the severe social burden it causes, has garnered significant attention from AI researchers. Numerous auxiliary diagnostic models have been developed with the aim of improving AD diagnostic services and thereby reducing the social burden. However, due to a lack of validation regarding the clinical value of these models, no AD diagnostic model has been widely accepted by clinicians or officially approved for use in enhancing AD diagnostic services. The clinical value of traditional medical devices is validated through rigorous randomized controlled trials to prove their impact on clinical outcomes. In contrast, current AD diagnostic models are only validated based on their accuracy, and the relationship between these models and patient outcomes remains unknown. This gap has hindered the acceptance and clinical use of AD diagnostic models by healthcare professionals. To address this issue, we introduce the COADBench, a benchmark centered on clinical outcomes for evaluating the clinical value of AD diagnostic models. COADBench curated subjects from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database who have at least two cognitive score records (the most commonly used clinical endpoint in AD clinical trials) from different follow-up visits. To the best of our knowledge, for the first time, it links the cognitive scores of subjects with model performance, using patient cognitive scores as clinical outcomes after intervention to evaluate the models. Through the benchmarking of current mainstream AD diagnostic algorithms using COADBench, we find that there was no significant correlation between the subjects' cognitive improvement and the model's performance, which means that the current performance evaluation criteria of mainstream AD diagnostic algorithms are not combined with clinical value.

1. Introduction

Alzheimer's disease is the most common type of dementia, accounting for the largest proportion of dementia, because of its irreversible, high cost of diagnosis, no cure and other characteristics, to society has brought a very serious burden. In order to reduce the diagnostic cost and improve the diagnostic effect, artificial intelligence (AI) researchers have developed various deep learning models to assist the diagnosis of Alzheimer's disease. For example, *Qiu et al.* [1] use a multi-modal input model based on 3D CNN to make three classifications of subjects, and the best model achieves an AUC of 0.971; *Xing et al.* [2] use a binary classification of subjects based on dynamic images and a pre-trained CNN model, and the best model achieves an AUC of 0.95.

Alzheimer's disease currently lacks a cure, so the main purpose of diagnosis is to identify patients with reversible or delayed symptoms for treatment, improving clinical outcomes and thus benefiting patients. The clinical assessment of the effectiveness of Alzheimer's disease diagnosis is mainly based on the calculation of benefits (such as cognitive improvement) based on changes in clinical endpoints or alternative endpoints. However, the evaluation indicators (Accuracy, AUC, etc.) of the current AI models used to diagnose Alzheimer's disease are not directly or indirectly related to clinical value. This means that although an AI model achieves a high value of AUC in the diagnostic task of categorizing or multicategorizing subjects (normal, mild cognitive impairment, Alzheimer's), the clinical value based on patient benefit does not necessarily improve. For example, *Zhang et al.* [3] use a fusion input model based on 3D CNN and Transformer to binary classify subjects. The accuracy of the best model reaches 0.929, but the index of cognitive improvement of patients in clinical practice is only 0.806.

Currently, in other areas where AI models have been introduced, the correlation between model evaluation and clinical outcomes is low. *Tyler et al.* [4] propose an algorithm based on KNN-DSS to provide

[*] Corresponding author.
E-mail address: liuwenjing@stu.gxnu.edu.cn (W. Liu).

https://doi.org/10.1016/j.tbench.2025.100198
Received 24 October 2024; Accepted 1 March 2025
Available online 13 March 2025

weekly insulin injection recommendations for patients with type 1 diabetes (T1D), using the duration of time that the patient's HbA1c level remains within the safe range as the clinical outcome in conjunction with the algorithm; *Komorowski et al.* [5] propose an AI clinician who gives reinforcement learning to provide the best medical strategy to the patient, and use mortality rates to evaluate the AI doctor's medical strategy. *Adams et al.* [6] develop a sepsis alert system based on machine learning, deploy it in hospitals to monitor the situation of sepsis patients, and evaluated the performance of the system using in-hospital mortality as the clinical outcome of patients. But there is no comparable example of a model for diagnosing Alzheimer's disease. This can lead to high classification evaluation metrics such as AUC or Accuracy, but poor clinical outcomes. For example, when the model tends to accurately identify patients whose cognition cannot be improved, a high model accuracy does not result in improved clinical outcomes.

In order to solve the above problems, COADBench first considers the use of clinical outcomes to evaluate the diagnostic model of Alzheimer's disease.

In most current clinical trials, the endpoint of Alzheimer's disease is cognitive improvement, so cognitive improvement is a quantitative model and a clinically significant endpoint acceptable to experts [7–11]. Thus, we propose clinical benefit measures based on changes in patients' ADAS scores (which reflect patients' cognitive ability) during follow-up after diagnosis and treatment, which could be used to evaluate model performance.

Second, we select samples from Alzheimer's Disease Neuroimaging Initiative (ADNI). The sample inclusion criteria: patients have at least two follow-up visits, in the form of 3D imaging data and demographic non-imaging data, with three categories of subjects: normal, mild cognitive impairment, and Alzheimer's disease.

Third, we build COADBench based on clinical benefit indicators and benchmark datasets, and conduct benchmark testing on mainstream Alzheimer's diagnosis models using the constructed COADBench. Our contributions are as follows:

- To the best of our knowledge, for the first time, we introduce ADAS scores as surrogate outcomes in the evaluation of an Alzheimer's disease model, correlating the model's performance with clinical value.
- To the best of our knowledge, with ADAS scores as the center, we construct the first clinically valuable benchmark for evaluating Alzheimer's disease models.
- The evaluation of current mainstream Alzheimer's disease models based on COADBench reveal that: (1) When classification evaluation indicators such as Accuracy and AUC are used to evaluate the model, the model with the best performance may not be the model with the highest clinical value; (2) There was no significant positive correlation between the classified evaluation indicators and clinical benefit indicators based on ADAS scores.

The paper is structured as follows. Section 2 describes the definition of the problem. Section 3 reviews recent research on diagnostic models for Alzheimer's disease. Section 4 covers COADBench in detail. Section 5 introduces the experimental results and analysis based on COADBench. Section 6 summarizes the findings.

2. Problem definition

2.1. Definition of the AD diagnosis problem

The AD diagnosis task in the current mainstream research is defined as a classification problem as follows:

$$min\left\{\mathbb{E}_{(x,y)\sim D_{Tr}}\alpha L\big(m(x),y\big)+\beta R(m)\right\} \tag{1}$$

Where D_{Tr} is the training set, The $L\big(m(x),y\big)$ indicates the loss at data point (x,y) with AD diagnosis model m, $R(m)$ indicates the regularization term of the model m. The coefficients α and β trade off these terms.

2.2. Clinical assessment of patient cognition

Clinically, the main way to enhance patient benefit is by improving the patient's cognitive function, which is quantified through the ADAS scores obtained from multiple follow-ups after treatment. Additionally, it is important to reduce the various losses caused by inaccurate diagnoses.

$$\begin{cases} max\left\{\sum_{D_{Te}} f\big(m(x)\big) * p * max\left\{0, A' - A\right\}\right\} \\ f\big(m(x)\big) = \begin{cases} 1, m(x) = 1 \\ 0, other \end{cases} \\ min\left\{L_{FPR}\right\} \\ L_{FPR} = \dfrac{FP}{FP+TN} \end{cases} \tag{2}$$

Where D_{Te} is the test set, $m(x)$ represents the prediction result of the model, and A and A' represent the ADAS score values of the patient at the current and next follow-up visits, respectively. p is equal to 1 when the model prediction is correct; otherwise, p is equal to 0.

L represents the psychological impact on non-AD subjects when they are misdiagnosed as AD patients, as well as the losses incurred from further medical consultations. Since this part is difficult to quantify, we use the model's False Positive Rate on the test set as a substitute. FP represents false positive rate and TN represents true negative rate.

3. Related work

To evaluate the effectiveness of a model in diagnosing a particular disease, it is necessary to ensure that its correct diagnostic predictions have a positive impact on patients. For example, *Komorowski et al.* [5] use a model to provide medication strategies for sepsis patients. They demonstrate the model's effectiveness by showing that the lowest mortality rates occurred in patients whose actual dosages matched the AI's recommendations. *Tyler et al.* [4] demonstrate the effectiveness of their model by showing that patients' blood sugar levels improved after adjusting the medication dosage according to the model's recommendations. *Arbabshirani et al.* [12] not only demonstrate the accuracy and specificity of their model in diagnosing intracranial hemorrhage but also highlight its clinical impact. The model successfully identify patients initially deemed to require only routine examinations, upgrading them to needing immediate examinations. Radiologists confirm that 64% of these upgraded patients indeed have intracranial hemorrhages, thereby proving the model's effectiveness. Because deep learning and similar technologies must ensure improved patient outcomes before being applied clinically, it is not sufficient to merely focus on increasing the accuracy of disease diagnosis models.

Since there are no treatments that can stop or reverse AD, existing medications may alleviate symptoms but are typically only effective in the early stages of the disease [13]. As a result, much research focuses on accurately identifying early-stage AD patients. The effectiveness of these models is often evaluated based on accuracy, a computer-based metric. However, when these models are applied clinically, it is essential to consider not only their accuracy but also whether early intervention, following a model's identification of an early AD patient, can improve actual patient outcomes. Currently, there are no prospective studies to validate this aspect. Most models are trained and tested using publicly available Alzheimer's disease datasets and evaluated based on metrics such as accuracy, sensitivity, precision, specificity, and F-measure. For example, studies by *Suk et al.* [14], *liu et al.* [15], *Martinez-Murcia et al.* [16], *Feng et al.* [17], *Raza et al.* [18] are all based on these publicly available datasets and performance metrics. In prospective studies on the effectiveness of drugs in improving patient symptoms [19], the impact on patients is typically assessed using the

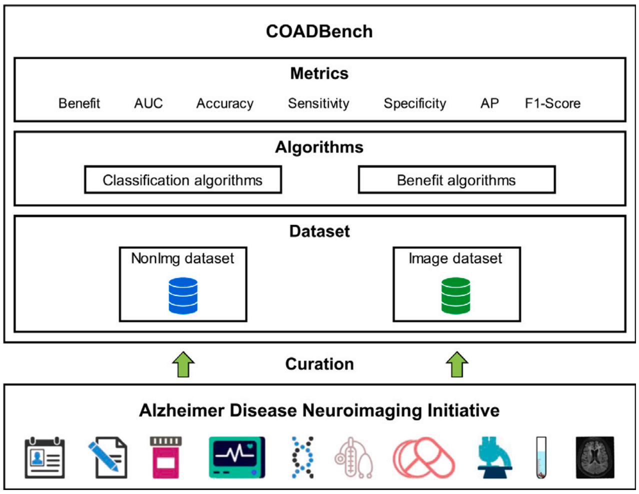


Fig. 1. The summary of COADBench benchmark framework.

Mini-Mental State Examination (MMSE) and the Alzheimer's Disease Assessment Scale-cognitive subscale (ADAS-cog).

In numerous medical domains, the assessment of model performance is frequently closely tied to actual patient outcomes. For example, in sepsis, which can result in rapid patient deterioration and death, model efficacy is often evaluated based on mortality rates. In contrast, AD remains incurable and progresses slowly [20], rendering mortality an impractical outcome measure. Consequently, current deep learning research on Alzheimer's disease focuses on early diagnosis, with model performance evaluation primarily relying on computational metrics such as accuracy, sensitivity, etc. However, reliance on these metrics alone is insufficient to demonstrate the model's positive impact on individual patients. Moreover, the absence of prospective studies further complicates the validation of the model's effectiveness in clinical practice.

4. COADBench

The structural block diagram of COADBench is shown in Fig. 1. The structural block diagram is viewed from bottom to top. Data from 13 types of medical examinations commonly used in AD diagnosis are selected from the Alzheimer's Disease Neuroimaging Initiative (ADNI) (http://adni.loni.usc.edu) and divided into non-imaging and imaging datasets to match different model inputs. AD diagnosis models generally use classification algorithms to diagnose subjects. In COADBench, we also use the benefit calculation algorithm to compute benefit metrics. For model evaluation, classification metrics (such as AUC, Accuracy, etc.) are used to assess the model's performance in classification, while benefit metrics are used to evaluate the model's clinical benefits for patients.

For model evaluation, classification algorithms and benefit calculation algorithms are used to obtain classification evaluation metrics (AUC, Accuracy, etc.) and benefit indicators, respectively.

4.1. Data sources

COADBench involves 10 tables and 3 categories of images which represent 13 categories of medical examinations data commonly used in AD diagnosis. The data are collected from 67 sites in the United States and Canada, contains 1543 subjects with 6225 visits, and all visits are labeled by one of three labels: AD (Alzheimer's disease), CN (Cognitively normal) and MCI (Mild cognitive impairment).

ADNI 13 kinds of medical tests shown in the list below:

Table 1
Characteristics of subjects.

		Number of subjects
Age	[55, 60)	39
	[60, 70)	311
	[70, 80)	790
	[80, 90)	391
	[90, 92]	12
Educate	[0, 3]	1525
	[4, 20]	9
Ethnic category	Hisp/Latino	46
	Not Hisp/Latino	1488
	Unknown	9
Racial category	White	1431
	More than one	14
	Black	64
	Asian	27
	Hawaiian/Other PI	1
	Unknown	6
Marriage	Married	1176
	Never married	53
	Widowed	178
	Divorced	130
	Unknown	6
Category	AD	330
	CN	408
	MCI	805

(1) Base information (Base), usually obtained through consultation, includes demographics, family history, medical history, and symptoms.
(2) Cognition information (Cog), usually obtained through consultation and testing, includes Alzheimer's Disease Assessment Scale, Mini-Mental State Exam, Montreal Cognitive Assessment, Clinical Dementia Rating, and Cognitive Change Index.
(3) Cognition testing (CE), usually obtained through testing, includes ANART, Boston Naming Test, Category Fluency-Animals, Clock Drawing Test, Logical Memory-Immediate Recall, Logical Memory-Delayed Recall, Rey Auditory Verbal Learning Test, Trail Making Test.
(4) Neuropsychiatric information (Neur), usually obtained through consultation, includes Geriatric Depression Scale, Neuropsychiatric Inventory, and Neuropsychiatric Inventory Questionnaire.
(5) Function and behavior information (FB), usually obtained through consultation, includes Function Assessment Question, Everyday Cognitive Participant Self Report, Everyday Cognition Study Partner Report.
(6) Physical neurological examination (PE), usually obtained through testing, includes Physical Characteristics, Vitals, and neurological examination.

The rest of the examinations include blood testing (Blood), urine testing (Urine), nuclear magnetic resonance scan (MRI), positron emission computed tomography scan with 18-FDG (FDG), positron emission computed tomography scan with AV45 (AV45), gene analysis (Gene), and cerebrospinal fluid analysis (CSF).

4.2. Benchmark datasets

To assess different AD diagnosis model, COADBench data source into two parts: image data and non-image data. The image data includes nuclear magnetic resonance scan imaging (MRI), positron emission computed tomography (PET) image, while the non-image data includes the remaining 10 types of tabular data from ADNI.

The demographic information of benchmark datasets subjects is shown in Table 1.

4.3. Classification algorithms

AD diagnosis models typically use classification algorithms, usually binary classification (normal individuals, Alzheimer's disease patients) or three-way classification (normal individuals, mild cognitive impairment, Alzheimer's disease patients). Based on the model's output format, the following classification methods are used:

- The model's output consists of a number of values ranging from [0, 1], corresponding to the number of classes, which represent the probabilities of the subjects belonging to each category. The category associated with the highest probability is then selected as the model's judgment result for the subject.
- The model's output is a score representing the subject's level of cognitive impairment. A threshold (in the case of binary classification) or two thresholds (in the case of three-way classification) are needed to map the score to a specific category. For three-way classification, for example, when the model outputs a COG_Score, thresholds a and b can be used to determine the specific category according to the following formula:

$$Category = \begin{cases} CN, COG\_Score \leq a \\ MCI, a < COG\_Score < b \\ AD, b \leq COG\_Score \end{cases} \tag{3}$$

4.4. Metrics

In COADBench, in addition to the common classification evaluation indicators such as AUC, Accuracy, Sensitivity, Specificity and AP, we also introduce the clinical indicator benefit to evaluate the benefit of a model to the subject. Benfit computation formula is as follows:

$$M = \sum_{i}^{n} l * b_i \tag{4}$$

$$B = \frac{1}{m} \sum_{i}^{n} l * p * b_i \tag{5}$$

Where l indicates the label of the subject (AD is 1 and others are 0), p indicates the prediction of the subject, If the cognition of the subject has not improved, then b=0, otherwise b is the difference between the subject's current ADAS-Cog and the follow-up ADAS-Cog.

Please note that all operations involving the subtraction of metrics in the paper assume that the difference between the two confidence intervals of the respective metrics is both independent and normally distributed.

5. Experimental results and analysis

COADBench is constructed based on the mainstream four Alzheimer's diagnosis models for the benchmark test. The benchmarking process for each diagnosis model is roughly the same, requiring data preprocessing, model training, and evaluation using both classification indices and image evaluation metrics.

5.1. Experimental setup

Experiments were conducted on a machine equipped with an NVIDIA A100 80 GB PCIe GPU, Intel Xeon Silver 4208 CPU, 256 GB RAM, and a 16TB HDD running CentOS 7.9. The hyperparameters of the experimental models are shown in Table 3.

Our process for evaluating AD diagnostic models is as follows: First, we save multiple intermediate models at different stages of training. For each intermediate model, we calculate classification evaluation metrics such as AUC and Accuracy, as well as the benefit metric on the test set.

After calculating the various metrics, the model with the highest AUC or Accuracy is selected as the one with the best classification performance, while the model with the highest benefit is considered the most beneficial for patients.

5.2. Data preprocessing

Each Alzheimer's diagnosis model the required data form is not the same, some model using only the image data, some model only using the image data, and some models use a mixed input of image and image data.

Image data preprocessing typically involves only standardizing the image size, while non-image data requires data cleaning. This includes removing features with too many missing values, removing features with excessive single-value entries, and filling in the missing values. The meanings of some of the main columns in the data are shown in Table 4.

Our benchmark data set of each record according to follow-up time and ADAS — cog difference to calculate the practice guideline values, so that the follow-up evaluation model of calculating the practice guideline values, the calculation formula of the practice of index in 4.4. The dataset was divided into training, validation, and test sets in a 6:2:2 ratio.

5.3. Model

We selected the following AD diagnostic models for evaluation:

- *Qiu et al.* [1] proposed three models to classify subjects into three categories: an MRI model based on 3D CNN and multilayer perceptron, using only MRI images as input; CatBoost based nonImg model uses only non-image data as input; the Fusion model based on CatBoost uses a mixed input of non-image data and image data. In Qiu et al.'s paper, the Fusion model finally achieved the best performance. We benchmarked for all three models.
- *Xing et al.* [2] used dynamic image-based and pre-trained CNN models to dichotomize subjects (AD vs CN). In Xing et al.'s paper, they used approximate rank pooling to convert 3D MRI into 2D dynamic image. The pre-trained CNN model was then input.
- *Zhang et al.* [3] use CNN and the Transformer based on 3 d model of the subjects for binary classification (AD vs CN), use only the image data as input.
- *Hosseini et al.* [21] proposed a deep 3D convolutional neural network for three-classification of subjects with Alzheimer's disease, using MRI images as input.

5.4. Results

The results of the evaluation of the mainstream AD diagnostic models are shown in Table 2. As can be seen from the table, when the AUC and other indicators of the AD diagnostic model reach their highest, the benefit value is not the highest in most of cases, which means that it is problematic to use AUC and other classification evaluation indicators to select the most effective model, because the model selected according to this method is not necessarily the most beneficial model for patients.

If we only look at the situation with the highest index, there is not much difference between the index value of the model with the best classification effect and the model that is most beneficial to patients, but not all AD diagnostic models can achieve good classification effects. For example, the Multimodal Nonimg model in Table 1 has the highest accuracy of 0.7619. The corresponding benefit is 0.8310, but when taking the model with the highest benefit, the benefit reaches 0.8886 when the accuracy is only 0.6577. This indicates that when the classification effect of the model is not very good, the index values of the model with the best classification effect and the model that is most beneficial to the patient may differ greatly.

In order to better analyze the experimental results and illustrate our point, we plot the scatter plot 2 with categorical metrics on the X-axis and benefit on the Y-axis. Each point in the scatter plot represents a

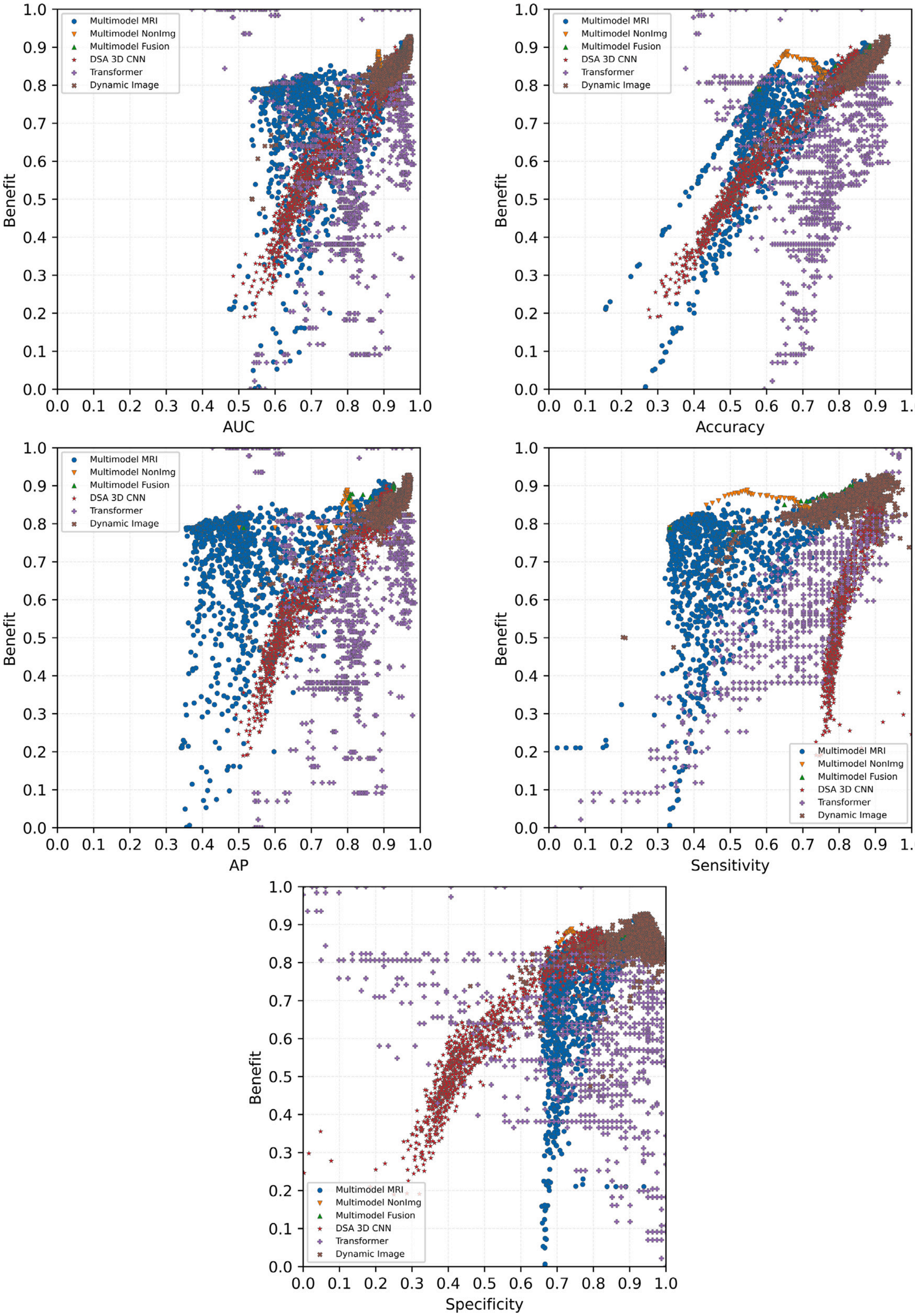


Fig. 2. Scatter plot of metrics versus benefit.

Table 2
Classification Metrics vs Benefit.

	best AUC		best Benefit		best Accuracy		best Benefit	
	AUC	Benefit	AUC	Benefit	Accuracy	Benefit	Accuracy	Benefit
Multimodal MRI [1]	0.9591	0.8926	0.9477	0.9119	0.8806	0.9020	0.8698	0.9119
Multimodal NonImg [1]	0.8978	0.8139	0.8846	0.8886	0.7619	0.8310	0.6577	0.8886
Multimodal Fusion [1]	0.9548	0.8770	0.9528	0.9051	0.8857	0.9045	0.8828	0.9051
DSA 3D CNN [21]	0.9477	0.8624	0.9307	0.9007	0.8596	0.8712	0.8314	0.9007
Transformer [3]	0.9830	0.5913	0.6008	0.9839	0.9638	0.8064	0.4275	0.9839
Dynamic Image [2]	0.9753	0.9180	0.9737	0.9286	0.9367	0.9079	0.9340	0.9286

	best AP		best Benefit		best Sensitivity		best Benefit	
	AP	Benefit	AP	Benefit	Sensitivity	Benefit	Sensitivity	Benefit
Multimodal MRI [1]	0.9186	0.8965	0.8863	0.9119	0.8595	0.9119	0.8595	0.9119
Multimodal NonImg [1]	0.8241	0.8128	0.7993	0.8886	0.7503	0.8173	0.5466	0.8886
Multimodal Fusion [1]	0.9280	0.8679	0.9255	0.9051	0.8518	0.9025	0.8386	0.9051
DSA 3D CNN [21]	0.9295	0.8624	0.9060	0.9007	0.9230	0.8518	0.8929	0.9007
Transformer [3]	0.9859	0.5913	0.6135	0.9839	0.9825	0.8225	0.9474	0.9839
Dynamic Image [2]	0.9748	0.9180	0.9702	0.9286	0.9926	0.7379	0.9213	0.9286

	best Specificity		best Benefit	
	Specificity	Benefit	Specificity	Benefit
Multimodal MRI [1]	0.9223	0.8905	0.9191	0.9119
Multimodal NonImg [1]	0.8488	0.8251	0.7404	0.8886
Multimodal Fusion [1]	0.9227	0.9045	0.9190	0.9051
DSA 3D CNN [21]	0.8459	0.8377	0.7673	0.9007
Transformer [3]	0.9877	0.7204	0.0617	0.9839
Dynamic Image [2]	0.9968	0.8224	0.9468	0.9286

Table 3
Model hyperparameters. Since the Multimodal NonImg and Multimodal Fusion models are based on the CatBoost regressor, there is no need to set batch size, optimizer, or loss function.

Model	Learning rate	Batch size	Epochs	Optimizer	Loss function
Multimodal MRI [1]	0.001	3	100	Adam	MSE
Multimodal NonImg [1]	0.05	–	100	–	–
Multimodal Fusion [1]	0.05	–	100	–	–
DSA 3D CNN [21]	0.000015	4	100	Adam	Cross entropy
Transformer [3]	0.0001	4	40	Adam	Cross entropy
Dynamic Image [2]	0.00001	16	100	Adam	Cross entropy

Table 4
Non-imaging data column meaning.

Column	Meaning
RID	Unique identifier of subject
VISCODE	Follow-up time
filename	The corresponding MRI file name
COG	Sample classification
Other	Feature

model with a different level of training, and the different shapes of the points distinguish between different model architectures.

From the trend of the scatter plot, it can be seen that when the classification evaluation metrics reach higher values, the benefit metrics are also high. This indicates that when the model performs well in classification, the benefits for AD patients are significant. However, when the classification evaluation metrics are not very high, there is not always a linear relationship between the classification metrics and the benefit metrics. All classification metrics of the DSA 3D CNN and Dynamic Image models show a clear positive correlation with the benefit metrics, particularly evident with the Accuracy metric. The Accuracy metric of the Multimodal MRI model also shows a certain positive correlation with the benefit metrics, while other models did not show this relationship. This implies that when the classification performance of a model did not reach a high level, one could not simply select the best classification model as the one that provides the highest benefit to patients. Focusing solely on classification performance during model training might overlook models that were truly beneficial to patients.

It is noteworthy that the Specificity metric of the Transformer model exhibited a tendency for a negative correlation with the benefit metric, which contrasts with other models. Furthermore, when the Specificity values of multiple intermediate models are similar, the benefit values can differ significantly. This may be due to the fact that Specificity reflects the model's classification accuracy for non-AD subjects, while the increase in benefit is related to AD subjects. When the model prioritizes identifying non-AD subjects and neglects the recognition of AD subjects, the benefit value tends to be lower. Conversely, high classification accuracy across all categories is necessary to achieve high values for both Specificity and benefit metrics. This further underscores the importance of not relying solely on classification evaluation metrics when selecting the most beneficial model for patients.

6. Conclusion

To the best of our knowledge, in this work, we are the first to associate AD (Alzheimer's Disease) diagnostic algorithms with clinical outcomes for evaluation, revealing the limitations of current mainstream AD algorithms and providing guidance for future development. However, our work have limitations. Due to challenges in clinical trials, we did not evaluate the algorithms in a real clinical environment but used cognitive improvement from clinical follow-ups as a proxy outcome, which may introduce bias. Additionally, our evaluation used data solely from the ADNI database, limiting patient diversity. To address these issues, we plan to create a hybrid evaluation system combining real-world and simulated data, expanding the scope to broader regions to reduce bias.

CRediT authorship contribution statement

Jiyue Xie: Data curation, Methodology, Resources, Validation, Visualization, Writing – original draft. **Wenjing Liu:** Supervision, Writing – original draft, Writing – review & editing. **Li Ma:** Writing – review & editing. **Caiqin Yao:** Writing – review & editing. **Qi Liang:** Writing – review & editing. **Suqin Tang:** Writing – review & editing. **Yunyou Huang:** Conceptualization, Formal analysis, Writing – review & editing.

Declaration of competing interest

The author Yunyou Huang is founding editor for BenchCouncil Transactions on Benchmarks, Standards and Evaluations and was not involved in the editorial review or the decision to publish this article. The other authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

We thank MGCS for providing the reference analysis of this work [22]. This work was supported by the National Natural Science Foundation of China (No. U21A20474 to S.T.).

References

[1] S. Qiu, M.I. Miller, P.S. Joshi, J.C. Lee, C. Xue, Y. Ni, Y. Wang, I. De Anda-Duran, P.H. Hwang, J.A. Cramer, et al., Multimodal deep learning for Alzheimer's disease dementia assessment, Nat. Commun. 13 (1) (2022) 3404.

[2] X. Xing, G. Liang, H. Blanton, M.U. Rafique, C. Wang, A.-L. Lin, N. Jacobs, Dynamic image for 3D MRI image Alzheimer's disease classification, in: European Conference on Computer Vision, Springer, 2020, pp. 355–364.

[3] Y. Zhang, K. Sun, Y. Liu, D. Shen, Transformer-based multimodal fusion for early diagnosis of Alzheimer's disease using structural MRI and PET, in: 2023 IEEE 20th International Symposium on Biomedical Imaging, ISBI, IEEE, 2023, pp. 1–5.

[4] N.S. Tyler, C.M. Mosquera-Lopez, L.M. Wilson, R.H. Dodier, D.L. Branigan, V.B. Gabo, F.H. Guillot, W.W. Hilts, J. El Youssef, J.R. Castle, et al., An artificial intelligence decision support system for the management of type 1 diabetes, Nat. Metab. 2 (7) (2020) 612–619.

[5] M. Komorowski, L.A. Celi, O. Badawi, A.C. Gordon, A.A. Faisal, The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care, Nature Med. 24 (11) (2018) 1716–1720.

[6] R. Adams, K.E. Henry, A. Sridharan, H. Soleimani, A. Zhan, N. Rawat, L. Johnson, D.N. Hager, S.E. Cosgrove, A. Markowski, et al., Prospective, multi-site study of patient outcomes after implementation of the TREWS machine learning-based early warning system for sepsis, Nature Med. 28 (7) (2022) 1455–1460.

[7] S. Gavrilova, I. Kolykhalov, N. Selezneva, Y. Kalyn, G. Jarikov, N. Mikhailova, A. Bratsoun, Clinical efficacy of exelon in patients with Alzheimer's disease, Eur. Neuropsychopharmacol. (9) (1999) 330–331.

[8] D.S. Geldmacher, Donepezil (aricept®) for treatment of Alzheimer's disease and other dementing conditions, Expert. Rev. Neurother. 4 (1) (2004) 5–16.

[9] G. Razay, G.K. Wilcock, Galantamine in Alzheimer's disease, Expert. Rev. Neurother. 8 (1) (2008) 9–17.

[10] F. Smith, Mixed-model analysis of incomplete longitudinal data from a high-dose trial of tacrine (cognex®) in Alzheimer's patients, J. Biopharm. Statist. 6 (1) (1996) 59–67.

[11] R. Wolz, A.J. Schwarz, K.R. Gray, P. Yu, D.L. Hill, Alzheimer's Disease Neuroimaging Initiative, et al., Enrichment of clinical trials in MCI due to AD using markers of amyloid and neurodegeneration, Neurology 87 (12) (2016) 1235–1241.

[12] M.R. Arbabshirani, B.K. Fornwalt, G.J. Mongelluzzo, J.D. Suever, B.D. Geise, A.A. Patel, G.J. Moore, Advanced machine learning in action: identification of intracranial hemorrhage on computed tomography scans of the head with clinical workflow integration, NPJ Digit. Med. 1 (1) (2018) 9.

[13] S. Fathi, M. Ahmadi, A. Dehnad, Early diagnosis of Alzheimer's disease based on deep learning: A systematic review, Comput. Biol. Med. 146 (2022) 105634.

[14] H.-I. Suk, S.-W. Lee, D. Shen, Alzheimer's Disease Neuroimaging Initiative, et al., Hierarchical feature representation and multimodal fusion with deep learning for AD/MCI diagnosis, NeuroImage 101 (2014) 569–582.

[15] S. Liu, S. Liu, W. Cai, H. Che, S. Pujol, R. Kikinis, D. Feng, M.J. Fulham, et al., Multimodal neuroimaging feature learning for multiclass diagnosis of Alzheimer's disease, IEEE Trans. Biomed. Eng. 62 (4) (2014) 1132–1140.

[16] F.J. Martinez-Murcia, A. Ortiz, J.-M. Gorriz, J. Ramirez, D. Castillo-Barnes, Studying the manifold structure of Alzheimer's disease: a deep learning approach using convolutional autoencoders, IEEE J. Biomed. Heal. Inform. 24 (1) (2019) 17–26.

[17] W. Feng, N.V. Halm-Lutterodt, H. Tang, A. Mecum, M.K. Mesregah, Y. Ma, H. Li, F. Zhang, Z. Wu, E. Yao, et al., Automated MRI-based deep learning model for detection of Alzheimer's disease process, Int. J. Neural Syst. 30 (06) (2020) 2050032.

[18] M. Raza, M. Awais, W. Ellahi, N. Aslam, H.X. Nguyen, H. Le-Minh, Diagnosis and monitoring of Alzheimer's patients using classical and deep learning techniques, Expert Syst. Appl. 136 (2019) 353–364.

[19] C. Wattmo, Å.K. Wallin, E. Londos, L. Minthon, Predictors of long-term cognitive outcome in Alzheimer's disease, Alzheimer's Res. Ther. 3 (2011) 1–13.

[20] R.A. Sperling, P.S. Aisen, L.A. Beckett, D.A. Bennett, S. Craft, A.M. Fagan, T. Iwatsubo, C.R. Jack Jr., J. Kaye, T.J. Montine, et al., Toward defining the preclinical stages of Alzheimer's disease: Recommendations from the National Institute on Aging-Alzheimer's Association workgroups on diagnostic guidelines for Alzheimer's disease, Alzheimer's Dement. 7 (3) (2011) 280–292.

[21] E. Hosseini-Asl, G. Gimel'farb, A. El-Baz, Alzheimer's disease diagnostics by a deeply supervised adaptable 3D convolutional network, 2016, arXiv preprint arXiv:1607.00556.

[22] Y. Huang, J. Zhao, D. Cui, Z. Yang, B. Xia, Q. Liang, W. Liu, L. Ma, S. Tang, T. Hao, et al., Quantifying the dynamics of harm caused by retracted research, 2024, arXiv preprint arXiv:2501.00473.

BenchCouncil Transactions on Benchmarks, Standards and Evaluations 4 (2024) 100199

Contents lists available at ScienceDirect

BenchCouncil Transactions on Benchmarks, Standards and Evaluations

journal homepage: www.keaipublishing.com/en/journals/benchcouncil-transactions-on-benchmarks-standards-and-evaluations/

Full length article

Evaluating long-term usage patterns of open source datasets: A citation network approach

Jiaheng Peng, Fanyu Han, Wei Wang [*]

School of Data Science and Engineering, East China Normal University, Shanghai, 200062, China
Engineering Research Center of Big Data Management, Shanghai, China
Engineering Research Center of Blockchain Data Management (East China Normal University), Ministry of Education, China

ARTICLE INFO

Keywords:
Dataset evaluation
Open source datasets
Usage Pattern Analysis
Citation network

ABSTRACT

The evaluation of datasets serves as a fundamental basis for tasks in evaluatology. Evaluating the usage patterns of datasets has a significant impact on the selection of appropriate datasets. Many renowned Open Source datasets are well-established and have not been updated for many years, yet they continue to be widely used by a large number of researchers. Due to this characteristic, conventional Open Source metrics (e.g., number of stars, issues, and activity) are insufficient for evaluating the long-term usage patterns based on log activity data from their GitHub repositories.

Researchers often encounter significant challenges in selecting appropriate datasets due to the lack of insight into how these datasets are being utilized. To address this challenge, this paper proposes establishing a connection between Open Source datasets and the citation networks of their corresponding academic papers. By mining the citation network of the corresponding academic paper, we can obtain rich graph-structured information, such as citation times, authors, and more. Utilizing this information, we can evaluate the long-term usage patterns of the associated Open Source dataset.

Furthermore, this paper conducts extensive experiments based on five major dataset categories (Texts, Images, Videos, Audio, Medical) to demonstrate that the proposed method effectively evaluates the long-term usage patterns of Open Source datasets. Additionally, the insights gained from the experimental results can serve as a valuable reference for future researchers in selecting appropriate datasets for their work.

1. Introduction

The evaluation of datasets is a cornerstone in various domains of research, forming a critical foundation for advancing the field of evaluatology [1]. High-quality datasets serve as essential building blocks for designing experiments, validating models, and deriving insights across disciplines [2]. As the volume of data and the diversity of datasets grow exponentially, the ability to evaluate and select appropriate datasets has become a vital skill for researchers [3]. Central to this process is the understanding of dataset usage patterns, which offer insights into their practical utility, relevance, and long-term significance [4]. However, this understanding is often obscured by the limitations of conventional evaluation metrics, particularly in the context of Open Source datasets.

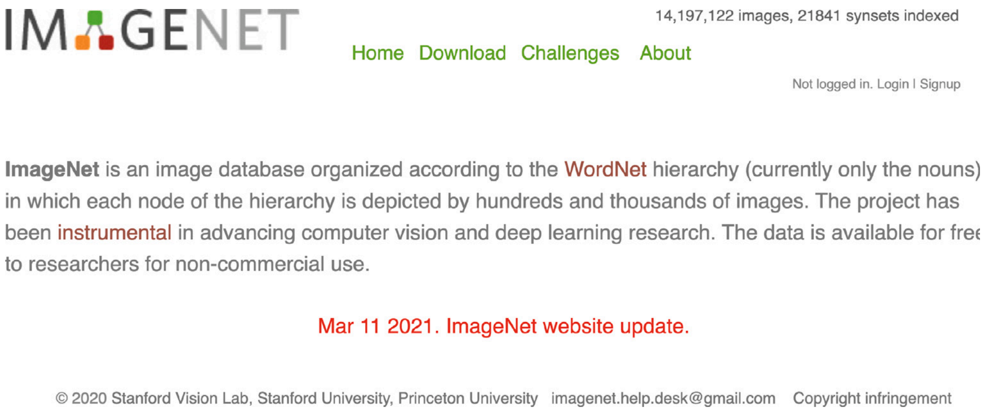
Open Source datasets have gained widespread attention for their accessibility, collaborative development, and impact on the research ecosystem. Notably, many renowned Open Source datasets maintain their prominence and continued usage over extended periods, even without frequent updates or maintenance. For example, Fig. 1 displays the official website of the well-known dataset ImageNet in the image processing domain. As shown in the figure, the website provides only limited information, such as a brief introduction to the dataset and download links. However, it does not offer any insights into the dataset's recent usage or updates. In contrast, Fig. 2 presents the corresponding GitHub repository for the ImageNet dataset. From this figure, it is evident that the repository has not been updated for over a year, suggesting that no significant activity has occurred during this period. This lack of recent logs or updates poses a challenge for us in understanding the dataset's current usage trends.

When researchers select datasets for data science tasks, their choices are often driven by personal subjective preferences, such as opting for well-known datasets they are familiar with. However, they lack factual evidence derived from behavioral log data to understand the recent and long-term usage patterns of these datasets.

Common data insight metrics are derived from the activity log data of GitHub repositories (e.g., stars, issues, forks, and activity levels on

[*] Correspondence to: School of Data Science and Engineering, East China Normal University, Shanghai, 200062, China.
E-mail address: wwang@dase.ecnu.edu.cn (W. Wang).

https://doi.org/10.1016/j.tbench.2025.100199
Received 6 January 2025; Received in revised form 14 March 2025; Accepted 19 March 2025
Available online 28 March 2025



IMAGENET

14,197,122 images, 21841 synsets indexed

Home Download Challenges About

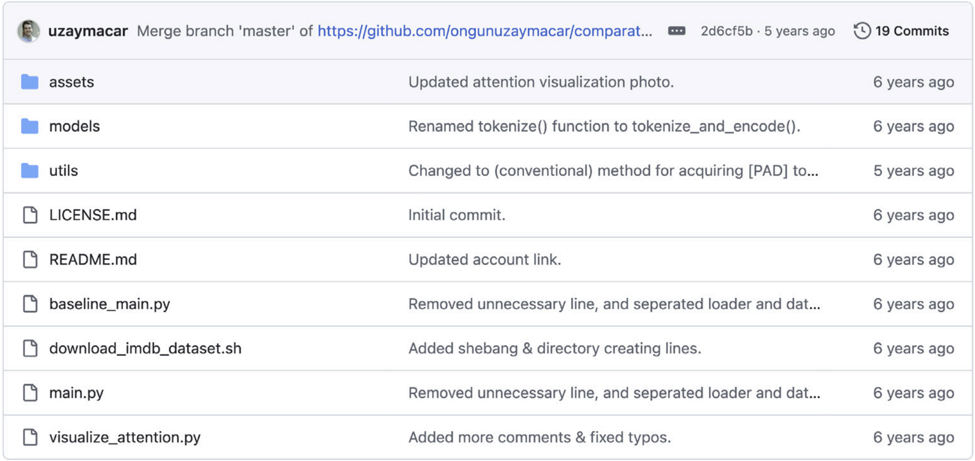
Not logged in. Login | Signup

ImageNet is an image database organized according to the WordNet hierarchy (currently only the nouns) in which each node of the hierarchy is depicted by hundreds and thousands of images. The project has been instrumental in advancing computer vision and deep learning research. The data is available for free to researchers for non-commercial use.

Mar 11 2021. ImageNet website update.

© 2020 Stanford Vision Lab, Stanford University, Princeton University imagenet.help.desk@gmail.com Copyright infringement

Fig. 1. ImageNet dataset official website.



uzaymacar Merge branch 'master' of https://github.com/ongunuzaymacar/comparat... 2d6cf5b · 5 years ago 19 Commits

assets	Updated attention visualization photo.	6 years ago
models	Renamed tokenize() function to tokenize_and_encode().	6 years ago
utils	Changed to (conventional) method for acquiring [PAD] to...	5 years ago
LICENSE.md	Initial commit.	6 years ago
README.md	Updated account link.	6 years ago
baseline_main.py	Removed unnecessary line, and seperated loader and dat...	6 years ago
download_imdb_dataset.sh	Added shebang & directory creating lines.	6 years ago
main.py	Removed unnecessary line, and seperated loader and dat...	6 years ago
visualize_attention.py	Added more comments & fixed typos.	6 years ago

Fig. 2. Github repository of the IMDB dataset.

GitHub repositories), which are used to measure the long-term popularity and developer activity of a repository. However, these metrics are heavily reliant on repository log activity data. In particular, when a repository has minimal log activity but its dataset continues to be widely used, these data insight metrics become ineffective.

For researchers, this gap presents a significant challenge. The lack of a comprehensive understanding of dataset usage patterns often results in inefficient selection processes and suboptimal utilization of resources. Without reliable indicators of long-term relevance and impact, researchers face difficulties in identifying datasets that best align with their specific needs and objectives. This limitation calls for innovative approaches to evaluate datasets that transcend traditional metrics and incorporate a more nuanced understanding of their role in the academic and research ecosystem.

In response to this challenge, this paper proposes a novel method to bridge the gap between Open Source datasets and their corresponding academic influences. We observed that most Open Source datasets are accompanied by a corresponding academic paper authored by the dataset's creators. This allows us to establish a connection between the dataset and the citation network of its associated academic paper.

Specifically, it establishes a connection between Open Source datasets and the citation networks of their associated academic papers. Academic papers often serve as a formal record of the development, application, and impact of datasets, and their citation networks offer a wealth of information. By mining and analyzing the citation networks, we can uncover critical data points such as citation counts, author contributions, collaboration patterns, and the influence of cited works. This approach leverages the inherent richness of graph-structured citation data to evaluate long-term usage patterns, providing a more comprehensive and reliable basis for dataset assessment.

This study conducts extensive experiments across five major categories of datasets — Texts, Images, Videos, Audio, and Medical — to validate the proposed approach. The experimental results demonstrate the effectiveness of utilizing citation network analysis for understanding the long-term usage and relevance of datasets. Insights derived from this evaluation not only contribute to the broader field of Open Source dataset assessment but also offer practical value to researchers. By enabling more informed decision-making in dataset selection, this work aims to improve the overall efficiency and impact of research efforts.

The contributions of this study are as follows:

- We propose an innovative approach that connects the GitHub repositories of Open Source datasets with the citation networks of their corresponding academic papers. Beyond addressing the direct challenges in existing dataset evaluation methods, this dual perspective enriches our understanding of the Open Source ecosystem. Furthermore, it provides a holistic framework for assessing datasets in a rapidly evolving research landscape, offering valuable insights into both their practical usage and academic influence over time.
- We not only analyze the usage patterns of Open Source datasets from a temporal perspective by examining citation timelines, but also explore potential collaboration patterns within the corresponding GitHub repositories by constructing various collaboration networks. These networks provide valuable insights into

the underlying reasons for the repository's development and sustained influence, shedding light on the factors driving its continued growth and relevance in the Open Source ecosystem. Open Source ecosystems and provides a holistic framework for evaluating datasets in a rapidly evolving research landscape.
- The findings presented in this work aspire to serve as a guide for researchers, dataset curators, and policymakers, fostering a deeper appreciation of the long-term value of Open Source datasets and their critical role in advancing scientific discovery.

2. Related works

The evaluation of Open Source datasets has attracted considerable attention in both academic and industrial domains, primarily due to the growing reliance on datasets for various tasks, including machine learning, data analytics, and scientific research. Existing studies on dataset evaluation can be broadly categorized into two areas: (1) methods and metrics for assessing Open Source projects and (2) citation network analysis for understanding academic influence and impact.

2.1. Open source project evaluation and dataset evaluation metrics

Metrics for evaluating Open Source projects often focus on repository-level statistics such as the number of stars, forks, issues, pull requests, and contributors. These metrics serve as proxies for popularity, community engagement, and activity levels. For example, there are tools and frameworks designed to provide insights into Open Source data, such as Open Source data insight integration plugins [5], mining collaborative patterns in Open Source communities [6], analyzing the geographical distribution of Open Source developers [7], and deriving insights from student performance in Open Source education programs [8], among others. However, when the target of analysis involves underlying collaboration networks, these tools and methods prove to be insufficient.

To address these limitations, researchers have explored more comprehensive graph-based frameworks for evaluating collaborative behaviors, such as Open Source maturity models and quality assurance metrics. For instance, influence assessment models based on contribution metrics, such as the OpenRank model [9], and collaboration pattern mining methods using OpenRank have been proposed [10]. While these models offer more effective ways to evaluate Open Source software, their applicability to dataset evaluation remains limited. This is primarily because these models lack sufficient data to capture the usage patterns and long-term relevance of datasets.

Several studies have proposed dataset-specific evaluation metrics, focusing on attributes like dataset size, diversity, annotation quality, and application domains. For instance, Schmidt et al. [11] highlighted the importance of dataset representativeness and its impact on model generalization. Similarly, Lalor et al. [12] proposed metrics for assessing the fairness and bias in datasets. While these approaches provide valuable insights into dataset quality, they do not address the longitudinal aspect of dataset usage in the research community.

2.2. Connecting datasets with citation networks: Research gaps and contributions

Citation network analysis has emerged as a powerful tool for understanding the academic influence of papers and their associated datasets. Researchers such as McLaren and Bruner [13] and Van Eck and Waltman [14] have demonstrated the potential of citation networks in identifying influential works, mapping collaboration patterns, and studying knowledge dissemination. These studies highlight the richness of citation data, which includes not only citation counts but also relationships between authors, institutions, and research domains.

Recent works have also explored the application of graph-based methods to analyze citation networks [15]. For example, Cummings and Nassar [16] utilized graph neural networks (GNNs) to predict the impact of scientific papers based on their position in the citation graph. Similarly, Liu et al. [17] and He et al. [18] studied the temporal evolution of citation networks to identify emerging research trends. These approaches underscore the value of leveraging graph-structured data to gain deeper insights into academic influence and usage patterns.

While research on Open Source project evaluation and citation network analysis has been extensive, there is a noticeable gap in connecting Open Source datasets with the citation networks of their corresponding academic papers. To date, no systematic efforts have been made to bridge this connection. Building on the insights from these related works, this paper addresses the gap by proposing a novel approach that combines Open Source dataset evaluation with citation network analysis. By leveraging the rich, graph-structured information in citation networks, this method provides a more comprehensive evaluation of long-term usage patterns. Unlike traditional metrics, it accounts for the enduring influence of datasets, offering valuable insights for researchers and dataset curators alike.

3. Methodology

In this section, we present the methodological framework employed to establish a connection between Open Source datasets and the citation networks of their corresponding academic papers. The goal of this methodology is to analyze the long-term usage patterns of datasets based on the citation activities of the academic papers associated with those datasets. The overall framework is depicted in Fig. 3.

3.1. Paper corresponding to the dataset

The first stage of the framework involves identifying the academic papers that correspond to the selected Open Source datasets. To achieve this, we leverage the paperswithcode platform. This is a widely used platform for collecting and organizing datasets along with their corresponding academic papers. We utilized this platform to obtain relevant data on Open Source datasets. The process can be divided into the following steps:

- Top-5 Selection: Using the API provided by the paperswithcode platform,[1] we retrieved the top five most popular dataset modality categories: text, image, video, audio, and medical. These five distinct modalities were selected to ensure comprehensive coverage across various data types and to capture diverse usage patterns in different research domains.
- Categorization: From each of these five modality categories, we selected representative datasets of small, medium, and large scales to ensure a balanced evaluation across different dataset sizes.
- Dataset Name Extraction: After categorization, we extract the names of the corresponding datasets. These dataset names are used as input for the next stage, which involves searching for the associated academic papers.

3.2. Citation network mining

The second stage of the framework focuses on mining the citation networks underlying their corresponding academic publications. This process is critical for evaluating the long-term impact and usage of the datasets. The following steps outline this process:

Searching via Semantic Scholar: We utilized the Semantic Scholar API[2]—an extensive academic search engine—to obtain the unique IDs of the corresponding papers by searching for the titles of the academic

[1] https://paperswithcode.com/

[2] https://api.semanticscholar.org/api-docs/graph

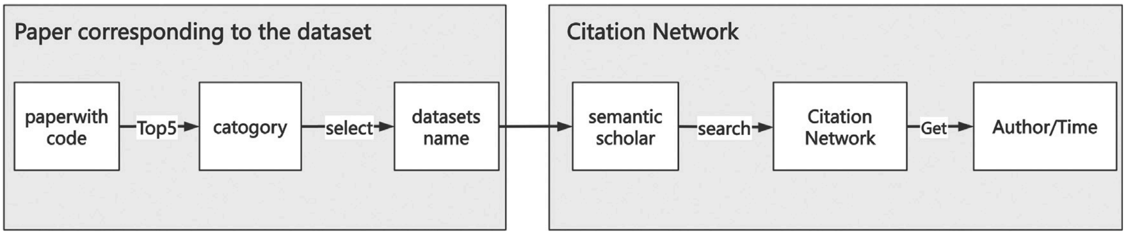


Fig. 3. Framework.

papers associated with each dataset. Using these IDs, we were able to mine the underlying citation networks and the networks of cited papers through the API. Subsequently, we retrieved all papers that cited the target papers and extracted relevant information from these citing papers.

Information Extraction: From the citation network, we extract key information, including:

- Title: The title of the cited papers.
- Authors: The authors of the cited papers and their affiliations.
- Publication Time: The publication date of the cited papers.

Below is a portion of the key code for citation network information mining using the Semantic Scholar API:

Algorithm 1 The method of citation network information mining using the Semantic Scholar API.

```
Input: paper title: $title$, optional fields: $fields$, paper ID: $paper\_id$, output file name: $output\_file$.
Output: Citation data CSV file.
$base\_url$ ← "https://api.semanticscholar.org/graph/v1/paper/search/match"
$params$ ← {"query": $title$, "fields": $fields$}
$response$ ← requests.get($base\_url$, $params = params$, $timeout = 10$)
$paper\_data$ ← $response.json()$
$paper\_id$ ← $paper\_data.get$("paperId")
$base\_url$ ← "https://api.semanticscholar.org/graph/v1/paper/" + $paper\_id$ + "/citations"
$citations$ ← []
$offset$ ← 0
while True do
    $params$ ← {"offset": $offset$, "limit": 1000, "fields": $fields$}
    $response$ ← requests.get($base\_url$, $params = params$, $timeout = 10$)
    $data$ ← $response.json()$
    $citations.extend(data.get$("data", []))
    if $data.get$("next") then
        $offset$ ← $data.get$("next")
    else
        break
    end if
end while
with open($output\_file$, "w", newline="", encoding="utf-8") as $file$:
    $writer$ ← csv.writer($file$)
    $writer.writerow$(["Paper Title", "Authors", "Publication Year"])
```

3.3. Evaluating long-term usage patterns

Through citation network information mining, we can leverage the obtained data to assess the long-term usage patterns of Open Source datasets.

By combining the dataset information with the citation network data, we can evaluate the long-term usage patterns of the selected Open Source datasets. The citation network provides a graph-structured representation of how the dataset's corresponding paper has influenced subsequent research over time. This approach addresses the limitations of conventional Open Source metrics by focusing on citation trends rather than repository activity alone.

Key Insights from the Framework:

- **Cumulative Citation Trend**: The total number of citations accumulated by a paper since its publication, calculated on an annual basis. This metric provides a historical perspective on the impact of the dataset-associated academic papers.
- **Annual Citation Growth Trend**: Refers to the number of new citations a paper receives each year since its publication.
- **Growth Rate Trend**: Also known as the growth speed, it represents the ratio of the increase in a data indicator to the base period data over a certain period, expressed as a percentage. This can be formulated as: $Y = \frac{X_t - X_{t-1}}{X_{t-1}} \times 100\%$ where Y denotes the growth rate, X_t and X_{t-1} represent the total number of citations in year t and t_{t-1},respectively.
- **Three Types of Collaborative Network Analysis**: Project Contribution Network analysis, Project Ecosystem Network analysis and Project Community Network analysis, all constructed via the Open Source project osgraph.[3]

  Project Contribution Network analysis: Find core project contributors based on developer activity information (Issues, PRs, Commits, CRs, etc.).

  Project Ecosystem Network analysis: Extract relationships between projects' development activities and organizations to build core project ecosystem relationships.

  Project Community Network analysis: Extract core developer community distribution based on project development activities and developer organization information.

In addition to analyzing academic papers associated with datasets that have been published for a considerable duration, the analyses of the **Annual Citation Growth Trend** and **Growth Rate Trend** also enable a clearer identification of datasets with substantial growth potential. This is particularly crucial for relatively new datasets that have been published for only one to three years, as their cumulative citation counts are typically lower. Citation networks not only provide information on the quantity and timing of dataset citations but also reveal the collaborative network structures formed around the datasets within the academic and industrial communities.

4. Experiment

4.1. Setup and datasets

For our study, we selected five distinct data modalities. Within each modality type, we established three different dataset scales. From each scale within every modality, we randomly selected one dataset to serve as the representative for that particular category and scale. Specifically, we included datasets of varying scales within each modality type — small, medium, and large. A small-scale dataset is defined as one

[3] https://github.com/TuGraph-family/OSGraph

Table 1
The specific names and categories of the selected datasets.

Category	Small-dataset	Medium-dataset	Large-dataset
Images	CityFlow (350)	Food-101 (2003)	Fashion-MNIST (7949)
Texts	FinQA (213)	CommonsenseQA (1349)	GLUE (6334)
Videos	MSVD (115)	OTB (2898)	UCF101 (5629)
Audio	XD-Violence (245)	Common Voice (1319)	Librispeech (5752)
Medical	VerSe (203)	ChestX-ray14 (2157)	MIMIC-III (6449)

Table 2
The selected dataset (with the total citation count of its corresponding academic paper).

Category	Small-dataset	Large-dataset
Images	JFT-3B (961) CityFlow (350) WildDeepfake (330)	CelebA (7959) Fashion-MNIST (7949) SVHN (6571)
Texts	CLINC150 (489) COCO (486) FinQA (213)	SST (8113) SQuAD (7686) GLUE (6334)

whose corresponding academic paper has fewer than 500 citations, a medium-scale dataset is defined as one with 500 to 5,000 citations of its corresponding paper, and a large-scale dataset is defined as one whose corresponding paper has been cited more than 5,000 times.

The specific dataset names corresponding to the five selected categories are presented in Table 1.

In addition, to conduct more comprehensive experiments and analyses, we randomly selected three datasets from each of the two domains (image and text), covering both large-scale and small-scale categories. The selected dataset names, along with their corresponding citation counts from the literature, are presented in Table 2.

4.2. Academic papers corresponding to the dataset

Table 1 and Table 2 lists the abbreviated names of each dataset. Below is a detailed description of their corresponding academic paper titles:

Fashion-MNIST [19]: A Novel Image Dataset for Benchmarking Machine Learning Algorithms

Food-101 [20]: Mining Discriminative Components with Random Forests

CityFlow [21]: A City-Scale Benchmark for Multi-Target Multi-Camera Vehicle Tracking and Re-Identification

GLUE [22]: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding

CommonsenseQA [23]: A Question Answering Challenge Targeting Commonsense Knowledge

FinQA [24]: A Dataset of Numerical Reasoning over Financial Data

UCF101 [25]: A Dataset of 101 Human Actions Classes From Videos in The Wild

OTB [26]: Object Tracking Benchmark

MSVD [27]: Collecting Highly Parallel Data for Paraphrase Evaluation

Librispeech [28]: An ASR corpus based on public domain audio books

Common Voice [29]: A Massively-Multilingual Speech Corpus

XD-Violence [30]: Not only Look, but also Listen: Learning Multimodal Violence Detection under Weak Supervision

MIMIC-III [31]: MIMIC-III, a freely accessible critical care database

ChestX-ray14 [32]: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases

VerSe [33]: A Vertebrae Labeling and Segmentation Benchmark for Multi-detector CT Images

JFT-3B [34]: Scaling Vision Transformers

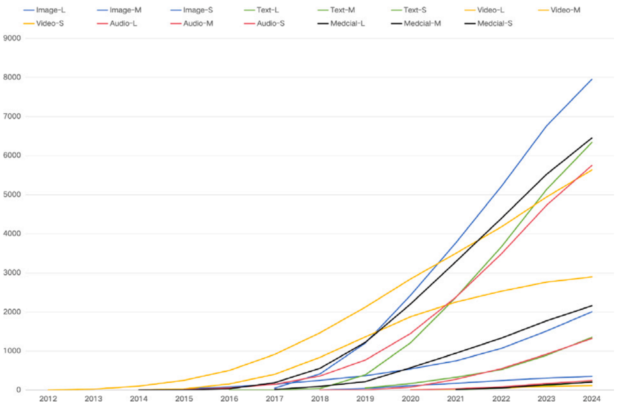


Fig. 4. The number of citations about datasets.

WildDeepfake [35]: A Challenging Real-World Dataset for Deepfake Detection.

CelebA [36]: Deep Learning Face Attributes in the Wild

SVHN [37]: Reading Digits in Natural Images with Unsupervised Feature Learning

SST [38]: Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank

CLINC150 [39]: An Evaluation Dataset for Intent Classification and Out-of-Scope Prediction

COCO-Text [40]: COCO-Text: Dataset and Benchmark for Text Detection and Recognition in Natural Images

4.3. The development status of datasets across different categories

Fig. 4 depicts the temporal evolution of cumulative citations for papers associated with datasets across various domains. From the perspective of cumulative citations, Image-L also stands out as the most prominent category. Its total citations have grown exponentially since 2017, far surpassing other categories by 2024. The datasets in the image domain have implicitly demonstrated their status as the most popular and highly scrutinized research area within the broader landscape of deep learning. Similarly, the cumulative citations of Text-L and Medical-L have also risen rapidly, particularly Text-L, whose growth trajectory has almost paralleled that of Image-L since 2020. This indicates that, in addition to the image domain, datasets in the text domain are also one of the focal points of researchers' attention.

In contrast, datasets in the video and audio domains (including large, medium, and small datasets) have seen slower growth in cumulative citations. Although Video-L and Audio-L have shown year-over-year increases in total citations, they still lag significantly behind the image and text domains. This may be due to the higher complexity of data processing and the more specialized application scenarios in these fields.

Overall, the trend in cumulative citations aligns with the trend in annual citation growth, where large-scale datasets — particularly in the image and text domains — continue to dominate, while medium and small-scale datasets, as well as those in the audio and video domains, have relatively lower influence and slower growth rates.

The annual citation growth trend is illustrated in Fig. 5. The annual growth trend in dataset citations clearly demonstrates the dominance of large-scale datasets. In particular, Image-L has seen a rapid increase in citations since 2016, peaking in 2022, followed by a slight decline in 2023 and 2024, while still maintaining the highest number of citations. This suggests that large-scale image datasets continue to attract significant attention from researchers and developers, despite the slowing growth in recent years.

Since 2017, the citation counts of most datasets have exhibited significant growth, particularly for Image-M and Text-M. This surge

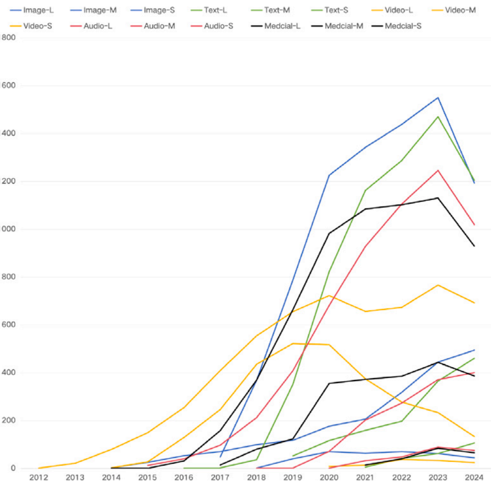


Fig. 5. The number of citations about datasets each year.

is likely attributable to the rapid development and widespread application of deep learning technologies during this period. Given that deep learning tasks in the image and text domains are among the most popular, the citation counts for datasets in these two fields have increased markedly.

Between 2020 and 2022, the growth rate peaked for most datasets, with the annual increase reaching its zenith in 2022. This peak may be associated with the heightened demand for datasets during the pandemic, as many studies shifted towards remote data collection and analysis. The increased reliance on datasets during this period likely contributed to the surge in citation counts.

Compared to large-scale datasets, medium-scale datasets exhibited less pronounced growth, possibly due to their narrower scope of applicability. The relatively slow development trend of small-scale datasets may be partly attributed to their limited application range and niche task suitability.

4.4. Evolutionary trends: Large- and small-scale datasets in image and text domains

The Fig. 6, Fig. 7 and Fig. 8 illustrate the development trends of image datasets of different scales (large and small) over the years, providing insights into their respective growth trajectories in terms of cumulative citations and annual citations.

4.4.1. Cumulative citation trends

In Fig. 6, the blue and red bars in the figure represent large-scale datasets, while the green and orange bars correspond to small-scale datasets. large-scale datasets (Image-L and Text-L) exhibit a significantly steeper growth curve compared to small-scale (Image-S and Text-S) dataset. The Fig. 6 illustrates the trends in cumulative citations for large-scale and small-scale datasets within the domains of image and text.

Around 2017, the number of citations for large-scale datasets began to increase rapidly. We posit that this surge is likely associated with the burgeoning development of deep learning. During this period, a significant number of researchers initiated work related to deep learning, which in turn led to a substantial increase in the citation counts of corresponding papers.

Beginning in 2017, the citation gap between large-scale datasets and small-scale datasets has progressively widened. By 2024, the highest citation count among the selected small-scale datasets was approximately 1000, with others exhibiting even lower citation frequencies. This trend suggests that small-scale datasets have not demonstrated robust capabilities in disseminating academic influence.

Through systematic observation and analysis, we have identified that this phenomenon can be primarily attributed to the fact that a substantial proportion of papers associated with large-scale datasets are published in top-tier conferences, particularly in premier venues such as CVPR, ICCV within the computer vision domains. These prestigious conferences, recognized as CCF-A class or Core Conference Ranking A* category, possess significant academic influence and visibility, thereby attracting greater attention from the research community and consequently generating higher citation rates.

4.4.2. Annual citation growth trends

As shown in Fig. 7, the blue and red bars in the figure represent large-scale datasets, while the green and orange bars correspond to small-scale datasets. large-scale datasets (Image-L and Text-L) exhibit a significantly steeper growth curve compared to small-scale (Image-S and Text-S) datasets. This exponential growth in cumulative citations for Image-L began around 2017. The analysis indicates that by 2020, large-scale datasets demonstrated an annual citation growth of about 1000 citations, far outpacing the growth observed in small-scale datasets. This substantial absolute increase underscores the continuing prominence and research value of large-scale image datasets, primarily due to their fundamental contributions to multiple deep learning sub-fields such as image recognition, object detection, and text classification.

Statistical evidence indicates that large-scale datasets often attain remarkable research impact, as reflected in their citation metrics, within the first three years after publication. On the contrary, small-scale datasets exhibited a markedly slower trajectory in citation growth. Typically, even after several years of availability, their cumulative citation counts remained within the range of a few hundred citations.

Consistent with the findings in Section 4.4.1, we observe that papers associated with small-scale datasets are often not published in the most prestigious academic conferences or journals. Additionally, the deep learning tasks corresponding to these datasets tend to be relatively niche, with fewer researchers engaged in related work. Consequently, the growth in citation counts for these datasets is relatively slow.

4.4.3. Growth rate trends

In addition, we conducted experiments on the annual growth rate trends of the papers corresponding to these datasets, as shown in Fig. 8. Similar to Section 4.4.1 and Section 4.4.2, the blue and red bars in the figure represent large-scale datasets, while the green and orange bars correspond to small-scale datasets. The large-scale datasets (Image-L and Text-L) exhibit a significantly steeper growth curve compared to the small-scale datasets (Image-S and Text-S).

Given that the citation growth rate of papers typically experiences an explosive increase shortly after publication — reaching up to 3700% in some cases — we truncated growth rates exceeding 200% to ensure clarity in the trend visualization. This truncation represents the “explosive growth” phase, primarily to focus on the citation growth patterns after the initial surge in popularity. This approach allows us to observe the citation dynamics once the initial fervor surrounding the publication has subsided.

As can be observed from the figure, large-scale datasets typically experience a prolonged period of “explosive growth”, during which their citation counts increase rapidly. After this initial surge, the citation growth rate tends to decline gradually, yet remains relatively high. In fact, even eight to ten years after publication, the annual citation growth rate for these large-scale datasets can still exceed 20%.

In contrast, small-scale datasets exhibit a much shorter period of growth. They generally attract significant attention only in the first one to three years following publication. Their citation growth rates decline sharply thereafter, typically falling below 20% within five to seven years.

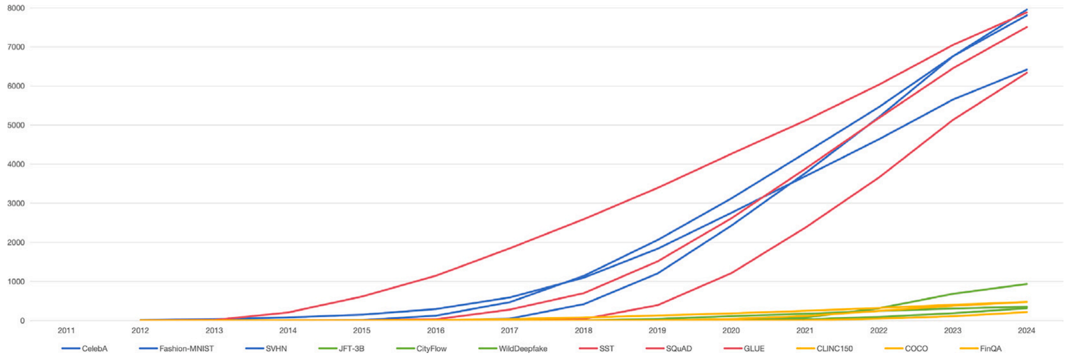


Fig. 6. Cumulative Citation Trends Across Image and Text Datasets.

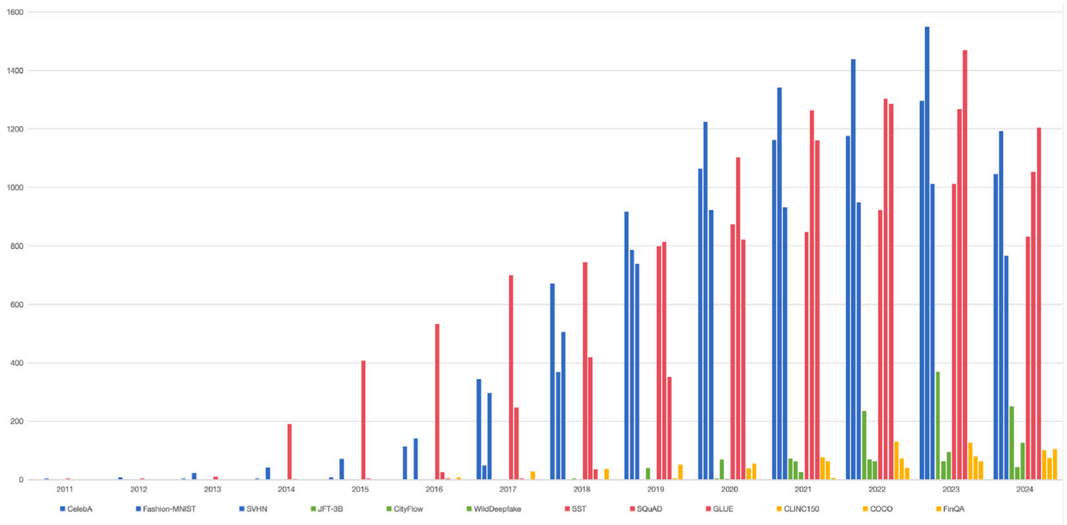


Fig. 7. Cumulative Citation Trends Across Image and Text Datasets Over the Years.

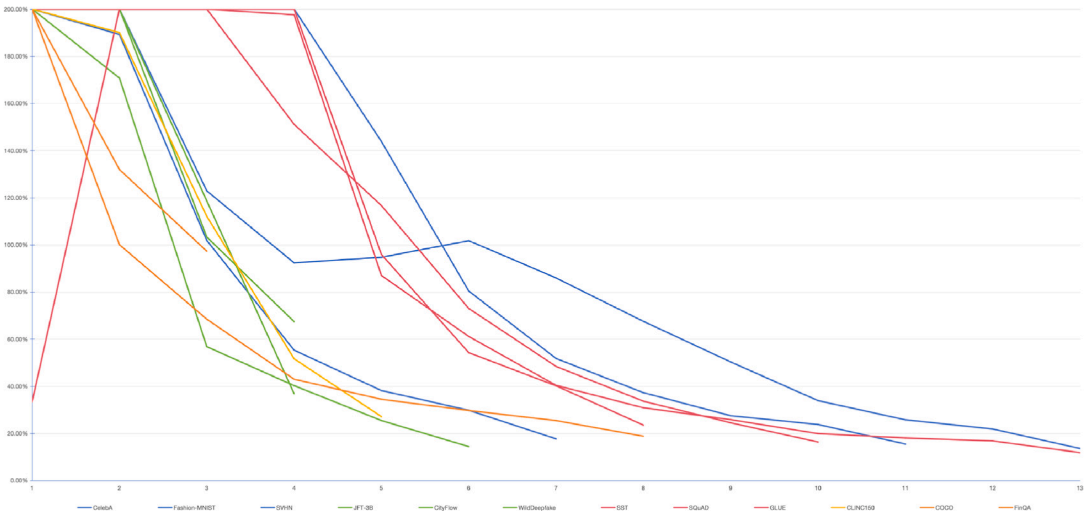


Fig. 8. Cumulative Citation Trends Across Image and Text Datasets.

4.5. Three types of collaborative network analysis

4.5.1. Project contribution network analysis

Taking the GitHub repository with the highest star count corresponding to a large-scale image dataset as an example, we conducted an in-depth analysis of its contribution collaboration network. The results reveal a highly active and diverse contributor network, comprising both individual contributors and automated bots. The visualization illustrates that the dataset has attracted numerous influential contributors, such as MarkDaoust, nealwu, and cshtjn, who have made significant contributions to the project through code reviews (CR), pull requests (PR), and issue discussions. These core contributors demonstrate the sustained interest and involvement of experienced developers in the ongoing maintenance and improvement of the dataset. (see Fig. 9).

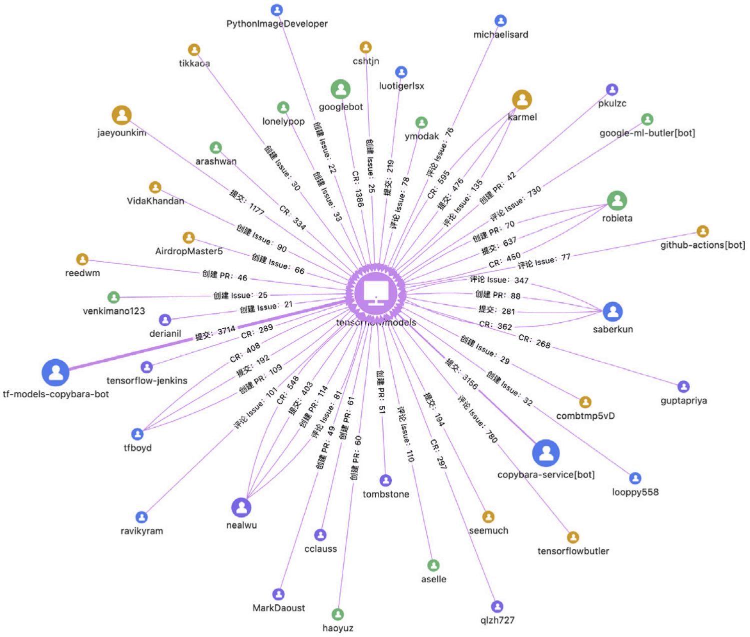


Fig. 9. Project Contribution Network.

In addition to individual contributors, the network also highlights the role of automated bots, including googlebot and tensorflowbutler, which represent well-known automated tools from major companies such as Google. These bots play a crucial role in maintaining the repository's automated workflows, indicating the importance of continuous integration/continuous deployment (CI/CD) processes in the project's lifecycle. The presence of such automation tools suggests that the repository maintains high standards of quality control, ensuring that updates and contributions are systematically reviewed and integrated.

Furthermore, the collaboration network demonstrates the involvement of a wide range of contributors across different organizational backgrounds, indicating the broad adoption and community-driven nature of the project. The combination of human contributors and automated bots highlights the hybrid nature of modern Open Source collaborations, where manual contributions are complemented by automated processes to ensure efficiency and reliability. This analysis underscores the significance of collaborative networks in maintaining large-scale Open Source datasets and the critical role of automation in facilitating seamless collaboration across distributed teams.

4.5.2. *Project ecosystem network analysis*

The project ecosystem collaboration network of the GitHub repository corresponding to a large-scale image dataset illustrates the extensive collaboration between this repository and other well-known projects in the Open Source community. As shown in the visualization, the repository attracts collaborations with several prominent repositories and organizations, including PyTorch, Microsoft's VS Code, and the Hugging Face community, along with its widely used Transformers library. These collaborations highlight the interconnectedness of major Open Source projects and demonstrate the dataset's influence across various domains of machine learning and software development.(see Fig. 10)

The network also reflects the growing importance of ecosystem-level interactions within Open Source communities. For instance, repositories such as TensorFlow, Keras, and Apache MXNet exhibit strong collaboration links with the dataset's repository, indicating shared contributions, joint development efforts, or the use of the dataset in complementary tools and frameworks. Such ecosystem interactions reinforce the dataset's role as a critical component within the broader machine learning infrastructure.

A particularly noteworthy observation is the emergence of the "rich-get-richer" effect, often referred to as the "rich club" phenomenon in network theory. The more a dataset or repository is cited and referenced within the community, the more likely it is to attract collaborations with other high-profile projects. This positive feedback loop results in widely-used datasets forming core hubs within the Open Source ecosystem, drawing further attention and engagement from influential developers and repositories. This effect underscores the importance of visibility and reputation in Open Source projects, where well-established repositories tend to attract more collaborators and maintain their central position within the ecosystem over time.

4.5.3. *Project community network analysis*

The community collaboration network of the GitHub repository corresponding to a large-scale image dataset with the highest star count demonstrates the extensive and diverse collaborations established with developers, companies, and research institutions across the globe. The network highlights significant contributions from developers and communities in countries such as China, Germany, United Kingdom, United States, and India, indicating the dataset's broad international adoption and its appeal to a wide range of contributors. (see Fig. 11)

Furthermore, the network reveals collaborations with some of the most prominent tech companies in the world, including Google, NVIDIA, and Microsoft, which play a crucial role in the development and promotion of cutting-edge machine learning technologies. The involvement of such well-established organizations suggests that the dataset is not only academically relevant but also practically significant for industry use cases. These collaborations reflect the repository's central position within the global Open Source ecosystem and its influence on both academic research and industrial applications.

This global and multi-organizational collaboration network underscores the growing importance of cross-border and cross-institutional partnerships in Open Source projects. The network demonstrates that widely-used datasets attract contributions from a diverse set of stakeholders, including independent developers, research institutions, and large technology companies. This diversity contributes to the repository's sustainability and long-term relevance by ensuring continuous improvements and the integration of new features driven by both academic and industry needs.

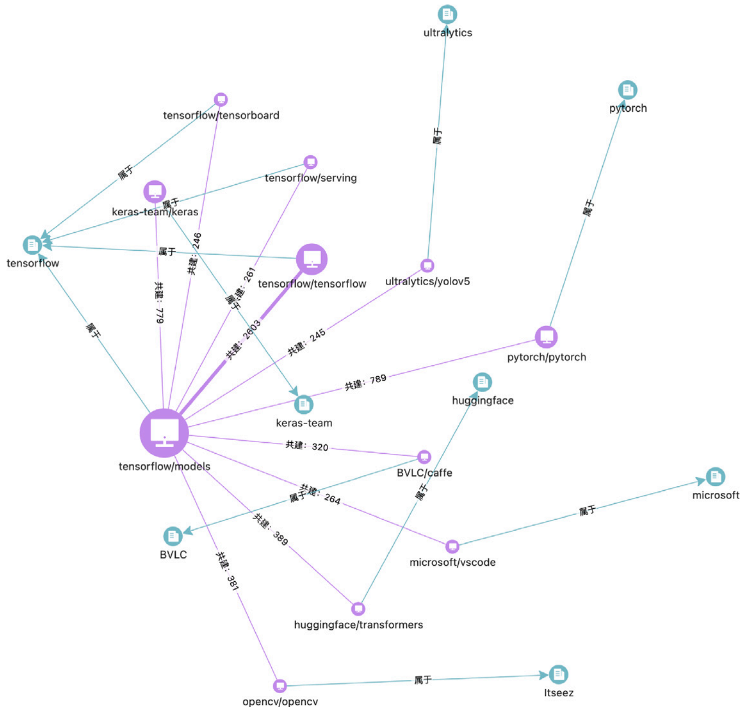


Fig. 10. Project Ecosystem Network.

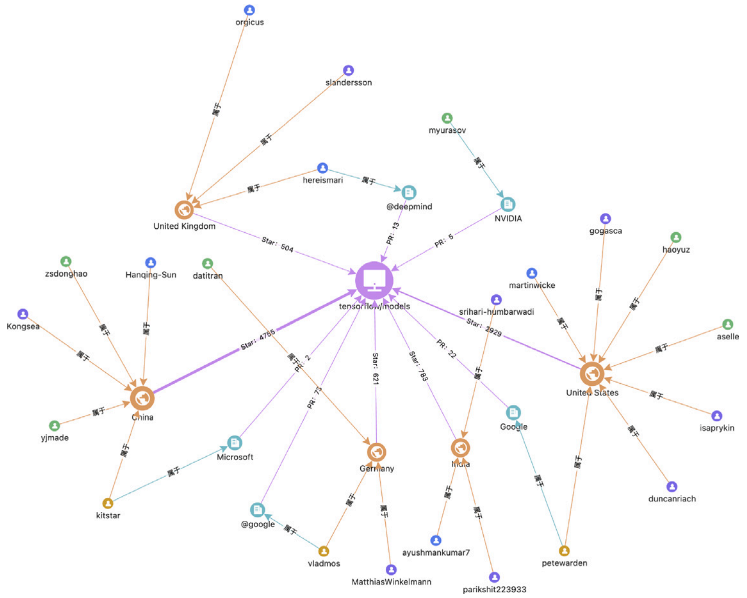


Fig. 11. Project Community Network.

Overall, the analysis of the project's community collaboration network highlights how large-scale datasets serve as focal points for global collaboration in Open Source ecosystems, driving innovation and knowledge sharing across countries and sectors.

5. Conclusion

This study proposes a novel framework for evaluating the long-term usage patterns of Open Source datasets by connecting them with the citation networks of their corresponding academic papers. Traditional data insight metrics, such as star counts and issue counts, become ineffective in the absence of GitHub log data. By mining the academic networks associated with datasets, we can indirectly analyze the long-term usage patterns of these datasets.

Through extensive experiments across five dataset modalities — text, image, video, audio, and medical — the study validates the effectiveness of the proposed method. The analysis of project contribution networks, ecosystem networks, and community networks reveals the collaborative nature of Open Source development and highlights the critical role of automated tools and global partnerships in sustaining large-scale repositories.

Overall, this research bridges the disconnect between Open Source activity metrics and academic citation analysis, laying the groundwork for a more holistic framework for assessing dataset relevance and impact. Nonetheless, several limitations remain in this study. For instance, some Open Source datasets lack corresponding published academic papers, or their associated papers have only recently been published, making citation information unavailable. In such cases, our approach is constrained. Addressing this issue is one of our future research directions. We aim to explore alternative methods to achieve a more comprehensive evaluation of Open Source datasets.

CRediT authorship contribution statement

Jiaheng Peng: Writing – original draft, Visualization, Software, Resources, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Fanyu Han:** Writing – review & editing, Investigation. **Wei Wang:** Project administration, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation grants 62137001, 62277017, and 61977026.

References

[1] J. Zhan, A short summary of evaluatology: The science and engineering of evaluation, BenchCouncil Trans. Benchmarks Stand. Eval. (2024) 100175.

[2] J. He, S. Yang, S. Yang, A. Kortylewski, X. Yuan, J.N. Chen, S. Liu, C. Yang, Q. Yu, A. Yuille, Partimagenet: A large, high-quality dataset of parts, in: European Conference on Computer Vision, Springer, 2022, pp. 128–145.

[3] N. Meron, V. Blass, G. Thoma, Selection of the most appropriate life cycle inventory dataset: new selection proxy methodology and case study application, Int. J. Life Cycle Assess. 25 (2020) 771–783.

[4] F.A. Silva, A.C. Domingues, T.R.B. Silva, Discovering mobile application usage patterns from a large-scale dataset, ACM Trans. Knowl. Discov. Data (TKDD) 12 (5) (2018) 1–36.

[5] Y. Tang, S. Zhao, X. Xia, F. Bi, W. Wang, HyperCRX: A browser extension for insights into GitHub projects and developers, in: Proceedings of the 32nd IEEE/ACM International Conference on Program Comprehension, 2024, pp. 460–464.

[6] W. Huang, X. Xia, A. Zhou, X. Zhou, W. Wang, S. Zhao, Z. Wang, S. Bian, OSGraph: A data visualization insight platform for open source community, in: International Conference on Database Systems for Advanced Applications, Springer, 2024, pp. 476–479.

[7] L. You, J. Peng, W. Wang, Y. Xia, K. Zhou, Data driven visualized analysis: Visualizing global trends of GitHub developers with fine-grained geo-details, in: International Conference on Database Systems for Advanced Applications, 2024, pp. 498–502.

[8] W. Jie, W. Huang, Z. Shengyu, X. Xiaoya, H. Fanyu, W. Wei, Y. Zhang, OpenRank contribution evaluation method and empirical study in open-source course, J. East China Norm. Univ. (Nat. Sci.) 2024 (5) (2024) 11.

[9] S. Zhao, X. Xia, B. Fitzgerald, X. Li, V. Lenarduzzi, D. Taibi, R. Wang, W. Wang, C. Tian, Motivating open source collaborations through social network evaluation: A gamification practice from Alibaba, 2023.

[10] S. Zhao, X. Xia, B. Fitzgerald, X. Li, V. Lenarduzzi, D. Taibi, R. Wang, W. Wang, C. Tian, OpenRank leaderboard: Motivating open source collaborations through social network evaluation in Alibaba, in: Proceedings of the 46th International Conference on Software Engineering: Software Engineering in Practice, 2024, pp. 346–357.

[11] L. Schmidt, S. Santurkar, D. Tsipras, K. Talwar, A. Madry, Adversarially robust generalization requires more data, Adv. Neural Inf. Process. Syst. 31 (2018).

[12] J.P. Lalor, A. Abbasi, K. Oketch, Y. Yang, N. Forsgren, Should fairness be a metric or a model? A model-based framework for assessing bias in machine learning pipelines, ACM Trans. Inf. Syst. 42 (4) (2024) 1–41.

[13] C.D. McLaren, M.W. Bruner, Citation network analysis, Int. Rev. Sport. Exerc. Psychol. 15 (1) (2022) 179–198.

[14] N.J. Van Eck, L. Waltman, CitNetExplorer: A new software tool for analyzing and visualizing citation networks, J. Informetr. 8 (4) (2014) 802–823.

[15] L. You, J. Peng, H. Jin, C. Claramunt, H. Zeng, Z. Zhang, DRGAT: Dual-relational graph attention networks for aspect-based sentiment classification, Inform. Sci. 668 (2024) 120531.

[16] D. Cummings, M. Nassar, Structured citation trend prediction using graph neural networks, in: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, IEEE, 2020, pp. 3897–3901.

[17] J. Liu, F. Xia, X. Feng, J. Ren, H. Liu, Deep graph learning for anomalous citation detection, IEEE Trans. Neural Netw. Learn. Syst. 33 (6) (2022) 2543–2557.

[18] G. He, Z. Xue, Z. Jiang, Y. Kang, S. Zhao, W. Lu, H2CGL: Modeling dynamics of citation network for impact prediction, Inf. Process. Manage. 60 (6) (2023) 103512.

[19] H. Xiao, K. Rasul, R. Vollgraf, Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017, arXiv preprint arXiv:1708.07747.

[20] L. Bossard, M. Guillaumin, L. Van Gool, Food-101–mining discriminative components with random forests, in: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13, Springer, 2014, pp. 446–461.

[21] Z. Tang, M. Naphade, M.Y. Liu, X. Yang, S. Birchfield, S. Wang, R. Kumar, D. Anastasiu, J.N. Hwang, Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 8797–8806.

[22] A. Wang, Glue: A multi-task benchmark and analysis platform for natural language understanding, 2018, arXiv preprint arXiv:1804.07461.

[23] A. Talmor, J. Herzig, N. Lourie, J. Berant, Commonsenseqa: A question answering challenge targeting commonsense knowledge, 2018, arXiv preprint arXiv:1811.00937.

[24] Z. Chen, W. Chen, C. Smiley, S. Shah, I. Borova, D. Langdon, R. Moussa, M. Beane, T.H. Huang, B.R. Routledge, et al., FinQA: A dataset of numerical reasoning over financial data, in: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, 2021, pp. 3697–3711.

[25] K. Soomro, UCF101: A dataset of 101 human actions classes from videos in the wild, 2012, arXiv preprint arXiv:1212.0402.

[26] A. Berg, J. Ahlberg, M. Felsberg, A thermal object tracking benchmark, in: 2015 12th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS, IEEE, 2015, pp. 1–6.

[27] D. Chen, W.B. Dolan, Collecting highly parallel data for paraphrase evaluation, in: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 2011, pp. 190–200.

[28] V. Panayotov, G. Chen, D. Povey, S. Khudanpur, Librispeech: an asr corpus based on public domain audio books, in: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, IEEE, 2015, pp. 5206–5210.

[29] R. Ardila, M. Branson, K. Davis, M. Kohler, J. Meyer, M. Henretty, R. Morais, L. Saunders, F. Tyers, G. Weber, Common voice: A massively-multilingual speech corpus, in: Proceedings of the Twelfth Language Resources and Evaluation Conference, 2020, pp. 4218–4222.

[30] P. Wu, J. Liu, Y. Shi, Y. Sun, F. Shao, Z. Wu, Z. Yang, Not only look, but also listen: Learning multimodal violence detection under weak supervision, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXX 16, Springer, 2020, pp. 322–339.

[31] A.E. Johnson, T.J. Pollard, L. Shen, L.w.H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Anthony Celi, R.G. Mark, MIMIC-III, a freely accessible critical care database, Sci. Data 3 (1) (2016) 1–9.

[32] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, R.M. Summers, Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2097–2106.

[33] A. Sekuboyina, M.E. Husseini, A. Bayat, M. Löffler, H. Liebl, H. Li, G. Tetteh, J. Kukačka, C. Payer, D. Štern, et al., VerSe: a vertebrae labelling and segmentation benchmark for multi-detector CT images, Med. Image Anal. 73 (2021) 102166.

[34] X. Zhai, A. Kolesnikov, N. Houlsby, L. Beyer, Scaling vision transformers, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 12104–12113.

[35] B. Zi, M. Chang, J. Chen, X. Ma, Y.G. Jiang, Wilddeepfake: A challenging real-world dataset for deepfake detection, in: Proceedings of the 28th ACM International Conference on Multimedia, 2020, pp. 2382–2390.

[36] Z. Liu, P. Luo, X. Wang, X. Tang, Deep learning face attributes in the wild, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 3730–3738.

[37] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, A.Y. Ng, et al., Reading digits in natural images with unsupervised feature learning, in: NIPS Workshop on Deep Learning and Unsupervised Feature Learning, vol. 2011, Granada, 2011, p. 4.

[38] R. Socher, A. Perelygin, J. Wu, J. Chuang, C.D. Manning, A.Y. Ng, C. Potts, Recursive deep models for semantic compositionality over a sentiment treebank, in: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, 2013, pp. 1631–1642.

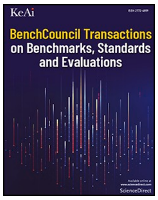
[39] S. Larson, A. Mahendran, J.J. Peper, C. Clarke, A. Lee, P. Hill, J.K. Kummerfeld, K. Leach, M.A. Laurenzano, L. Tang, et al., An evaluation dataset for intent classification and out-of-scope prediction, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP, 2019, pp. 1311–1316.

[40] A. Veit, T. Matera, L. Neumann, J. Matas, S. Belongie, Coco-text: Dataset and benchmark for text detection and recognition in natural images, 2016, arXiv preprint arXiv:1601.07140.

Contents lists available at ScienceDirect

BenchCouncil Transactions on Benchmarks, Standards and Evaluations

journal homepage: www.keaipublishing.com/en/journals/benchcouncil-transactions-onbenchmarks-standards-and-evaluations/



Full Length Article

Advanced Deep Learning Models for Improving Movie Rating Predictions: A Benchmarking Study

Manisha Valera [a], Dr. Rahul Mehta [b,*]

[a] *Research Scholar, Gujarat Technological University, Ahmedabad, India*

[b] *Department of Electronics & Communication Engineering, GEC Rajkot, Gujarat, India*

ARTICLE INFO

Keywords:
Variational autoencoder
Convolutional neural networks (CNN)
Recurrent neural networks (RNN)
Long short-term memory (LSTM)
gated recurrent unit (GRU)
Distilbert

ABSTRACT

Predicting movie ratings very precisely has become a vital aspect of personalized recommendation systems, which requires robust and high-performing models. for evaluating the effectiveness in predicting movie ratings, this study conducts a comprehensive performance analysis of various deep learning architectures, which includes BiLSTM, CNN + LSTM, CNN + GRU, CNN + Attention, CNN, VAE, Simple RNN, GRU + Attention, Transformer Encoder, FNN and ResNet. Here each model's performance is evaluated on movie reviews' dataset, enhanced with sentiment scores and user ratings, by using a range of evaluation metrics: Mean Absolute Error (MAE), R^2 score, Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Explained Variance. Here the results highlight distinct strengths and weaknesses among the models, in which VAE model consistently delivering superior accuracy, whereas attention-based models prove prominent improvements in interpretability and generalization. This analysis offers important insights into choosing models for movie recommendation systems, which also highlights the balance between prediction accuracy and computational efficiency. The discoveries from this study serve as a benchmark for future developments in movie rating prediction, supporting the researchers and practitioners in augmenting recommendation system performance.

1. Introduction

Despite noteworthy advancements in the field of recommendation systems, existing studies in this field still leave several important limitations unaddressed. Though traditional collaborative filtering methods are foundational, they frequently encounter challenges e.g., data sparsity and cold-start problems, which are particularly challenging to manage in movie recommendation systems. as observed in earlier research, models based purely on matrix factorization or basic recurrent architectures often struggle to capture the complex temporal patterns and emotional nuances that significantly influence user preferences. More advanced models which are using GRU and attention mechanisms, like those by Xia et al. [1] and Wang et al. [5], have improved in addressing these issues while considering time-based patterns. They still lack a full integration of sentiment analysis, which is really important for understanding user sentiments and likings toward movies.

In addition, multi-modal approaches that comprise data sources like movie posters and plot summaries, as demonstrated by Xia et al. [2] provide all-inclusive view of content preferences but lack robust sentiment-based personalization, which is crucial for the domains where emotional engagement is critical. Variational Autoencoders (VAEs), which are used effectively for collaborative filtering by Askari et al. [3] and Liang et al. [6], offer another trail for grasping hidden patterns in user interactions. However, these models tend to focus more on interaction data rather than user sentiment, possibly overlooking key insights that could improve recommendation accuracy and relevance. Furthermore, while sentiment-enhanced hybrid models have started to bridge this gap, as in Dang et al. [8], their incorporation remains limited, and the models face challenges in scalability and computational cost.

Existing literature, including Siet et al. [7], explores various architectures like CNNs, RNNs, and clustering-based methods, but lacks a comprehensive comparison across these models, limiting our understanding of their relative performance under a unified framework. Few studies provide a thorough evaluation of these models based on standardized error metrics, making it difficult to determine which approach consistently outperforms the others in terms of robustness, accuracy and recommendation quality.

To address these kinds of gaps, this paper provides a comprehensive comparison of state-of-the-art deep learning models for movie recommendation, including BiLSTM, CNN + LSTM, CNN + GRU, CNN +

* Corresponding author.
E-mail address: Rdmehta@hotmail.com (Dr.R. Mehta).

https://doi.org/10.1016/j.tbench.2025.100200
Received 10 December 2024; Received in revised form 23 March 2025; Accepted 11 April 2025
Available online 18 April 2025

Attention, CNN, VAE, Simple RNN, GRU + Attention, Transformer Encoder, FNN and ResNet. Exclusively, this work integrates sentiment analysis to enhance the models' ability to account for user emotions, adding a layer of personalization that prior models lacked. By evaluating each model on standardized error metrics—such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE)— this study aims to identify the most effective model for delivering accurate and sentiment-aware recommendations. Through this rigorous approach, our work sets a benchmark for future advancements in movie recommendation systems by emphasizing the impact of sentiment-driven personalization on recommendation quality.

2. Preliminaries

This paper presents a collaborative filtering recommendation algorithm [1] which integrates attention mechanisms within a Gated Recurrent Unit (GRU) framework and employs adversarial learning techniques. Here, the proposed model's aim is to enhance user-item interactions that focus on important features and reduce the noise from irrelevant data. In this, the results prove the enhanced performance in the context of recommendation accuracy compared to traditional methods, showcasing the effectiveness of the attention mechanism and adversarial learning in capturing user preferences.

This study proposes a multi-modal transformer framework [2] which leverages both textual and visual features from movie posters which are used to enhance the recommendation performance. In this, by employing an attention mechanism, the model's concentration is on prominent features from the posters while integrating them with textual data that is obtained from movie descriptions. The experimental results here illustrate that the proposed approach implicitly outperforms existing methods, mostly in scenarios where visual data plays a vital role in user preference prediction.

It explores the application of Variational Autoencoders (VAEs) [3] in the context of top-K recommendation systems using implicit feedback. In this the authors introduce an innovative VAE architecture that successfully models user preferences and item characteristics while also addressing challenges associated with implicit feedback, such as data sparsity. The experiments disclose that the proposed VAE-based method attains competitive results in top-K recommendation tasks, demonstrating its capability to generalize well to the hidden data.

This survey [4] provides an all-inclusive overview of various deep learning models, which includes Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Long Short-Term Memory networks (LSTMs), and Gated Recurrent Units (GRUs). In this the authors discuss the strengths and weaknesses of individual models, their applications in different domains, and comparative performance metrics. This paper contributes as a valuable resource for researchers and practitioners, for those who seek to understand the landscape of deep learning architectures.

In this work, the authors propose a personalized movie recommendation system [5] that combines LSTM and CNN architectures to capture both sequential and contextual features from user interactions. The model is designed here to extract temporal patterns in user behavior while considering the content of movies, as well. Experimental results indicate that the proposed system outperforms traditional collaborative filtering methods, particularly in capturing user preferences over time.

This paper investigates the use of Variational Autoencoders (VAEs) [6] for collaborative filtering tasks. The authors proposed a model here that collectively learns user and item representations while incorporating uncertainty into the recommendations. The VAE framework here addresses challenges as well, such as sparsity and cold-start problems in collaborative filtering, resulting in the enhanced recommendation accuracy. The discoveries suggest that VAEs can efficiently model user-item interactions in collaborative filtering scenarios.

This study focuses on improving movie recommendation systems [7] by integrating deep learning techniques and KMeans clustering. Here, the authors develop a sequence-based recommendation model that captures user preferences over time and apply KMeans to group similar users. The results demonstrate that the hybrid approach which yields superior recommendation performance compared to traditional methods, mostly in dealing with sequential data.

This paper explores the integration of sentiment analysis into recommender systems [8] which is used to enhance user experience. The authors propose a deep learning framework that incorporates sentiment scores from user reviews to refine recommendations, particularly for items with varying emotional tones. Here, the experiments show that incorporating sentiment analysis leads to more personalized and appropriate recommendations, highlighting the importance of emotional context in user preferences.

This work presents an approach [9] to improve movie recommendation systems by leveraging deep learning models alongside sentiment analysis. The authors demonstrate that incorporating sentiment data from user reviews significantly enhances the accuracy and relevance of recommendations. The findings underscore the potential of combining different data sources to create more effective recommendation algorithms. The SVM and CNN algorithms were implemented for the Movie Recommendation System to recommend the most relevant films for a given movie. Even after extensive testing, CNN classifier has produced decent findings in terms of suggesting the films.

This paper proposes the GANCF model [10], which combines user and item latent vectors with auxiliary information to enhance recommendation performance through deep non-linear learning. Here, experimental results show better outcomes on two datasets, validating the benefit of auxiliary data. Future work will discover time-based mechanisms and integrate multi-source heterogeneous data for better capturing dynamic user interests.

This study addresses the cold start [11] problem in recommendation systems and proposes a deep learning approach that builds user profiles from demographic attributes. Here, a modified ANN model clusters users by demographics, which is used to provide personalized movie recommendations. It also demonstrates strong performance across multiple evaluation metrics.

By human brain function, Convolutional Neural Networks (CNNs) are inspired [12,13] and They effectively handle grid-like structured data, too [14]. The CNN architecture has 3 types of dimensions: 1D is for processing text and signals, 2D is for images and audio, whereas 3D is for videos. While CNNs are chiefly used in computer vision tasks, e.g., image classification [15], they also perform well in text classification using word vectors formed through concatenation [16].

Google's TensorFlow framework [17], developed for machine learning, underlines tensors, which generalize vectors and matrices for managing flexible dimensions. This study leverages the Keras TensorFlow library which is mostly used to build CNN models with layers via including input, convolution, max-pooling, flatten, dense, dropout, and output [18], here each layer processes input data through the network [19]. Given the one-dimensional nature of text data, a 1D convolutional layer is used for the same, which is letting the model to extract composite patterns in the data [20].

To prevent overfitting, a dropout rate of 0.5 is applied after each Conv1D layer, deactivating a portion of neurons [21]. To reduce dimensionality, MaxPooling1D takes the maximum value in each pooling window. To further minimize the risk of overfitting, another dropout layer is added afterward MaxPooling. Then, a flattening layer converts the feature matrix into a vector, followed by another dropout at 0.5. Here, a dense layer with 64 units and ReLU activation processes this vector. The final dense layer with 3 units and sigmoid activation produces class probabilities. CNN was chosen for its aptitude to achieve greater accuracy and it effectively recognizes patterns, even in rating data.

The content-based recommendation system [22] is developed using CNN, which is combining TF-IDF and RoBERTa for pattern recognition in movie review data obtained from Twitter. This augmented CNN model with SMOTE and an SGD optimizer achieved an accuracy of 86.41 %, efficaciously providing accurate movie recommendations.

DistilBERT is a reorganized version of BERT, achieving a 40 % reduction in size and a 60 % increase in speed while conserving 97 % of BERT's language comprehension abilities [23]. It is trained through distillation, it's highly efficient and well-suited for edge deployments. Summarization can be extractive (selecting key sentences) or abstractive (rephrasing content). This study assesses BERT-based models and it introduces "SqueezeBERTSum," [24] which is a streamlined summarization model that retains 98 % of BERTSum's performance with 49 % fewer parameters. ArDBertSum, an Arabic text summarization model based on fine-tuned DistilBERT [25], enhanced with the SCSAR technique for sentence segmentation. It is evaluated on the EASC corpus, it beats other Arabic summarizers, and the future work will focus on expanding datasets, filtering evaluation methods, and discovering other pre-trained models.

This paper offers a widespread comparison of cutting-edge deep learning models for movie recommendation, encompassing BiLSTM, CNN + LSTM, CNN + GRU, CNN + Attention, CNN, VAE, Simple RNN, GRU + Attention, Transformer Encoder, FNN, and ResNet.

2.1. BiLSTM (Bidirectional long short-term memory)

BiLSTM networks are a variant of LSTMs [26] which process data in both directions - forward and backward, making them particularly effective in capturing long-range dependencies within sequences. In recommendation systems, BiLSTM is used to understand the consecutive patterns in user interactions (e.g., movie-watching behavior) over time. This bidirectional approach is helpful in capturing the complete context of user preferences, mainly for the tasks such as sequential recommendation.

2.2. CNN + LSTM

The CNN + LSTM model attaches Convolutional Neural Networks (CNNs) with LSTMs [5] which is allowing the system to handle both spatial and sequential data efficiently. CNNs are typically applied first to capture features from movie details (e.g., visual or textual features), followed by LSTMs to understand the chronological dependencies in user interaction data. This combination is powerful for multimedia recommendation systems where both temporal dynamics and content features (such as movie genres or user reviews) impact the movie recommendations.

2.3. CNN + GRU

This architecture pairs CNNs with Gated Recurrent Units (GRUs), where CNNs capture spatial features from input data, whereas GRUs manage sequential dependencies. The GRU, a simplified version of LSTM [10], combines the forget and input gates into an update gate and merges cell and hidden states. This kind of design allows it to capture long-term dependencies effectively while reducing issues like gradient vanishing and explosion. Here, GRUs are computationally more efficient than LSTMs, as they contain fewer parameters. The CNN + GRU model is thus appreciated for the cases where movie recommendation systems need to balance temporal insights with efficient processing of rich content data, like user comments / reviews.

2.4. CNN + attention

In this model, CNNs are coupled with an attention mechanism, which is used to prioritize important features in the data. First, CNNs extract core features, which are then weighted by the attention mechanism, allowing the model to focus on the utmost pertinent information. For movie recommendations, CNN + Attention mechanism can highlight detailed aspects of user preferences, such as genre or specific movie features, which is used here to provide more relevant suggestions based on past communications.

2.5. CNN (Convolutional neural network)

Originally developed for image processing, CNNs are proficient at recognizing spatial hierarchies in data. In context of movie recommendation systems, CNNs [27] are used for feature extraction from text-based reviews/ user comments, or visual features related to movie posters. Though CNNs do not inherently capture sequential information, they provide valuable acumens into content-related features that impact recommendations.

2.6. VAE (Variational autoencoder)

VAEs are probabilistic models, which are designed for dimensionality reduction and data generation. They use latent variable representations, which are particularly useful for collaborative filtering because they can capture the hidden factors which are driving user preferences. In movie recommendation systems, VAEs allow for the modeling of complex, implicit feedback, creating robust representations of user preferences that adapt well to various types of recommendation tasks.

2.7. Simple RNN (Recurrent neural network)

RNNs [4] are a foundational architecture which is used for sequential data processing, where each node's output is fed as input to the next node. While they are effective for short sequences, RNNs are prone to matters like vanishing gradients, which is limiting their ability to capture long-term dependencies. In context of recommendation task, Simple RNNs can offer elementary insights into sequential patterns. still, they are typically lacking in efficiency than more advanced recurrent models like GRUs or LSTMs.

2.8. GRU + attention

This model combines GRUs [4] with an attention layer because they want to prioritize significant sequential data. GRUs efficiently handle sequential dependencies as well, while the attention layer boosts interpretability by highlighting the most influential user interactions or content features. This kind of combination is ideal for recommendation systems that require efficient temporal modeling and the ability to focus on key preferences in the user's viewing history as well.

2.9. Transformer encoder

This is an element of the Transformer architecture that relies solely on self-attention mechanisms by discarding recurrence entirely. This kind of architecture allows parallel processing of input data, making it efficient and extremely scalable. Transformer Encoders are particularly effective in capturing complex dependencies in user interactions over time, making them suitable for large-scale recommendation systems and these systems need to analyze diverse content features simultaneously.

2.10. FNN (Feedforward neural network)

Feedforward Neural Networks are simple neural networks containing fully connected layers, mainly used for classification or regression tasks. In recommendation systems, FNNs can be beneficial for basic collaborative filtering tasks or as supplementary layers in hybrid models, though they lack the sequential or hierarchical structure required for complex, multi-faceted recommendation tasks.

2.11. ResNet (Residual neural network)

ResNet is DNN model which is known for its residual connections. it helps to mitigate issues like vanishing gradients in very deep networks. In recommendation contexts, ResNet can be applied to extract robust, hierarchical features from high-dimensional data, e.g., movie posters or other multimedia content. Its depth makes it especially effective for learning complex feature, contributing to high-quality recommendations.

3. DATASET preparation

Here, we present a movie recommendation system that integrates movie review datasets from numerous sources. This includes a dataset of over 5000 movies from data source Kaggle [29] up to year 2017, alongside movie metadata [30] and additional data from Wikipedia for movies released between the years 2018 [31], 2019 [32] and 2020 [33]. Additionally, we collect reviews for sentiment analysis from the TMDB website using the TMDB API [28].

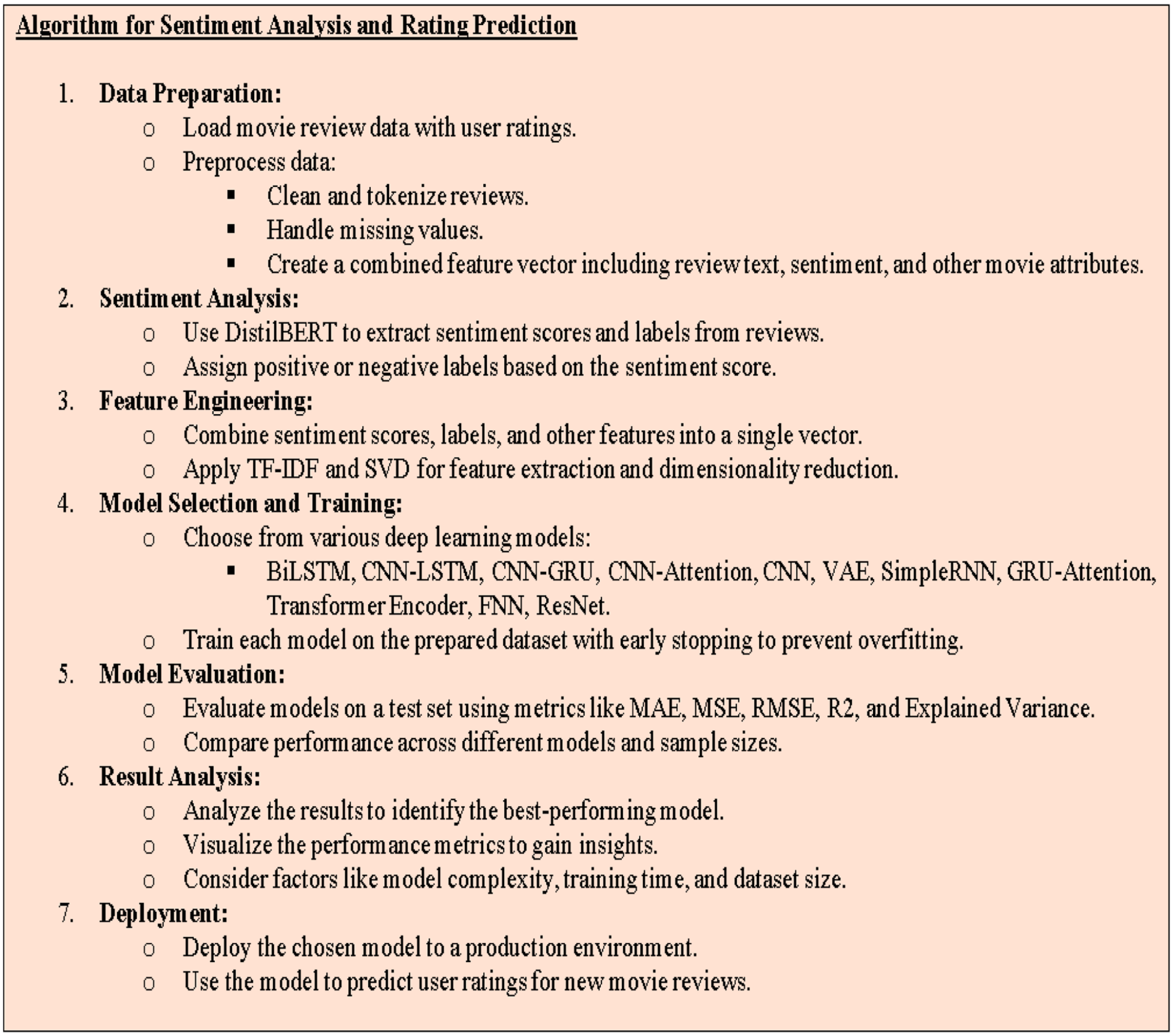
4. Feature engineering

In this project as shown in Fig. 1, the focus is on building an inclusive and sophisticated framework, for movie rating prediction by integrating various machine learning as well as deep learning models with sentiment analysis and feature extraction techniques. Initially, we preprocess the data using DistilBERT, which is a Transformer-based model used to extract sentiment scores and labels, enhancing the dataset with valuable contextual insights from movie reviews. Additionally, we apply TF-IDF vectorization, which is combined with SVD for the reduction of dimensionality, resulting in a streamlined feature set.

The experiments conducted in this study reveal the tangible impact of sentiment analysis on the performance of the classification model. Without sentiment analysis, the model struggled to generalize effectively, especially in handling imbalanced classes. However, integrating sentiment embeddings generated by DistilBERT led to a noticeable improvement in performance metrics, as detailed below.

The supplementary experiments on sentiment analysis, as shown in Table 1, demonstrate a clear improvement in both classification and regression tasks. The observed improvements in both classification and regression tasks suggest that sentiment embeddings contribute beyond sentiment polarity detection, directly enhancing rating prediction accuracy. While the classification accuracy increases from 85.75 % to 91.75 % with DistilBERT, demonstrating better sentiment differentiation, the key takeaway is its impact on regression performance. The reduction in MSE (from 0.1224 to 0.0743), MAE (from 0.2552 to 0.1595), and RMSE (from 0.3498 to 0.2726) underscore how refined sentiment representations lead to more precise numerical predictions.

Rather than treating classification accuracy as a standalone metric, it should be interpreted as a validation of sentiment embedding quality. Higher classification accuracy indicates that the embeddings capture



Algorithm for Sentiment Analysis and Rating Prediction

1. **Data Preparation:**
    - o Load movie review data with user ratings.
    - o Preprocess data:
        - ▪ Clean and tokenize reviews.
        - ▪ Handle missing values.
        - ▪ Create a combined feature vector including review text, sentiment, and other movie attributes.
2. **Sentiment Analysis:**
    - o Use DistilBERT to extract sentiment scores and labels from reviews.
    - o Assign positive or negative labels based on the sentiment score.
3. **Feature Engineering:**
    - o Combine sentiment scores, labels, and other features into a single vector.
    - o Apply TF-IDF and SVD for feature extraction and dimensionality reduction.
4. **Model Selection and Training:**
    - o Choose from various deep learning models:
        - ▪ BiLSTM, CNN-LSTM, CNN-GRU, CNN-Attention, CNN, VAE, SimpleRNN, GRU-Attention, Transformer Encoder, FNN, ResNet.
    - o Train each model on the prepared dataset with early stopping to prevent overfitting.
5. **Model Evaluation:**
    - o Evaluate models on a test set using metrics like MAE, MSE, RMSE, R2, and Explained Variance.
    - o Compare performance across different models and sample sizes.
6. **Result Analysis:**
    - o Analyze the results to identify the best-performing model.
    - o Visualize the performance metrics to gain insights.
    - o Consider factors like model complexity, training time, and dataset size.
7. **Deployment:**
    - o Deploy the chosen model to a production environment.
    - o Use the model to predict user ratings for new movie reviews.

Fig. 1. Algorithm of Sentiment Analysis & Movie rating Prediction.

Table 1
Performance comparison with and without DistilBERT sentiment classifier.

Model Variant	Accuracy	MSE	MAE	RMSE	Precision (Class 1)	Recall (Class 1)	F1-score (Class 1)
Without DistilBERT	0.8575	0.1224	0.2552	0.3498	0.86	1.00	0.92
With DistilBERT	0.9175	0.0743	0.1595	0.2726	0.91	1.00	0.95

nuanced sentiment variations more effectively, which, in turn, enrich the feature representations used in regression. This improved representation reduces prediction errors by aligning extracted sentiment information more closely with actual user ratings.

The CNN + LSTM model, trained with TF-IDF embeddings serves as a baseline to illustrate this relationship. While TF-IDF captures word frequency-based sentiment cues, DistilBERT embeddings offer a more contextualized understanding, leading to improvements across both classification and regression tasks Therefore, sentiment analysis should be framed primarily in terms of its role in refining feature extraction, ensuring consistency with the study's core regression evaluation metrics.

To enhance clarity, the discussion will emphasize how improvements in sentiment classification contribute to better rating predictions. This reinforces the alignment between sentiment analysis and the study's primary regression objectives.

5. MODEL development & results discussion

This dataset is then used to train multiple models encompassing BiLSTM, CNN + LSTM, CNN + GRU, CNN + Attention, CNN, VAE, Simple RNN, GRU + Attention, Transformer Encoder, FNN, and ResNet. Each model here explores different mechanisms for capturing dependencies within the data. For example, the CNN + Attention model utilizes self-attention which is used to identify relationships within the data, while the BiLSTM model captures dependencies which are in both forward & backward directions. Moreover, using a VAE-based model allows us to integrate generative elements, creating a more robust feature representation that can potentially improve prediction accuracy.

Here the evaluation criteria include MSE, MAE, RMSE, R-squared, and explained variance score, which help us to analyze model performance and provide insight into each model's suitability for the task. With this approach, our goal is to establish a strong baseline and identify the best-performing model, contributing to advanced movie recommendation systems that align closely with user preferences and actual ratings.

Here this pipeline allows for an inclusive assessment of innumerable deep learning models on the movie rating prediction task. It integrates both of the traditional architectures (like BiLSTM and CNN) and the advanced approaches (like Attention mechanisms, VAE, and GAN), with an emphasis on balancing performance and interpretability. Each model shown here is designed to address specific data characteristics, such as sequence information given in movie reviews, making the framework much flexible for various text-heavy recommendation systems.

Here's a summary of the models and their structures:

- BiLSTM: A Bidirectional LSTM network with 64 units, followed by a Dense layer (32 units) for regression. It uses MSE- Mean Squared Error as the loss function and Adam optimizer.
- CNN + LSTM: Combines Conv1D (64 filters) for feature extraction, followed by LSTM (64 units) for sequence modeling. It uses MSE and Adam optimizer.
- CNN + GRU: Similar to the CNN + LSTM, but replaces LSTM with GRU for sequence modeling. It also uses MSE and Adam.
- CNN + Attention: Uses Conv1D layers for feature extraction followed by a self-attention mechanism, then a Dense layer for regression. It uses MSE and Adam.
- VAE: A Variational Autoencoder with a 32-dimensional latent space. The model uses both reconstruction loss (binary cross entropy) and KL divergence loss, and is trained with the RMSprop optimizer.
- Simple RNN: A Simple RNN layer (64 units) is mainly used for sequence modeling, followed by a Dense layer for the regression. It uses MSE and Adam.
- GRU + Attention: This combines GRU (64 units) with self-attention for sequence modeling, followed by Dense layers for regression. It also uses MSE and Adam.
- FNN (Feedforward Neural Network): A fully connected network with three Dense layers of sizes 128, 64, and 32, trained for regression using MSE and Adam.
- ResNet: A CNN with residual connections and two Conv1D layers followed by MaxPooling, Flatten, and Dense layers for regression. It uses MSE and Adam.
- Transformer Encoder: A simplified transformer with Conv1D layers and Dense layers for regression. It uses MSE and Adam.
- GAN (Generator + Discriminator): The generator creates synthetic data from a latent vector, and the discriminator classifies the data. Both parts are trained using binary cross-entropy loss.

Based on the data provided from Table 2, here a summarized analysis of the models' observations and insights based on the sample size and key metrics (MAE, MSE, RMSE, R^2, and Explained Variance) is provided in detail:

5.1. Effect of attention mechanism

Models incorporating Attention (e.g., CNN + Attention, GRU + Attention) demonstrate varying degrees of improvement over their non-attention counterparts, particularly in reducing RMSE and improving R^2 for larger sample sizes.

However, CNN + Attention and GRU + Attention do not constantly outperform simpler models on smaller datasets, which may point to a need for larger data volumes to fully leverage the benefits of attention.

5.2. Performance of simple and advanced models

From analyzing Figs. 2, 3 and 4, As the sample size rises, The BiLSTM model shows consistent performance improvement, with comparatively lower MAE, MSE, and RMSE to other models. For the larger datasets (e.g., 5000 samples), it performs fairly well with R^2 values around 0.26.

As Per Fig. 5, Traditional models such as FNN (Feedforward Neural Networks) and ResNet have high error rates and poor R^2 values, mainly on smaller datasets, indicating they are less suited for this regression task without additional optimization.

More advanced models, such as the Transformer Encoder, demonstrate potential but generally fall behind BiLSTM and VAE in terms of MAE and RMSE across most sample sizes.

The performance analysis of various models reveals significant variations, highlighting the strengths and limitations of each approach. The VAE model consistently achieves the lowest error values (MAE, MSE, and RMSE), indicating its ability to capture complex patterns effectively. However, its highly negative R^2 and Explained Variance scores suggest potential overfitting or difficulties in generalizing to unseen data. This suggests that while VAE is powerful in latent representation learning, it may require additional regularization techniques or fine-tuning for better generalization. In contrast, BiLSTM demonstrates relatively strong performance with lower errors and improved R^2 scores, making it

Table 2
The evaluation metrics for given Different Models on customized dataset.

Sample Size	Model	MAE	MSE	RMSE	R2	Explained Variance
1000	BiLSTM	0.671496	0.979911	0.989904	−0.0009	0.00753
1000	CNN + LSTM	6.080096	37.88085	6.154742	−37.6921	−0.00128
1000	CNN + GRU	6.024232	37.1923	6.098549	−36.9888	−0.00059
1000	CNN + Attention	3.740283	14.59443	3.820265	−13.907	−0.02247
1000	CNN	1.80834	3.964826	1.991187	−3.04974	−0.05592
1000	VAE	0.406633	0.17754	0.421355	−2642.08	−101.288
1000	Simple RNN	1.64833	4.494392	2.119998	−3.59065	−3.44333
1000	GRU + Attention	1.46563	2.814831	1.677746	−1.87512	0.000344
1000	Transformer Encoder	0.717499	1.079262	1.038875	−0.10238	−0.09497
1000	FNN	5.951155	36.31172	6.02592	−36.0894	−0.00824
1000	ResNet	2.510871	8.314544	2.883495	−7.49262	−1.15139
2000	BiLSTM	0.59105	0.584029	0.764218	0.173399	0.198478
2000	CNN + LSTM	0.746794	0.899824	0.94859	−0.27356	0.000646
2000	CNN + GRU	5.175412	27.42779	5.237155	−37.8197	0.000728
2000	CNN + Attention	2.48067	6.941842	2.634738	−8.82508	−0.11759
2000	CNN	2.046236	4.959599	2.227016	−6.01953	−0.12288
2000	VAE	0.343646	0.129785	0.360257	−1833.7	−112.103
2000	Simple RNN	0.975142	1.555495	1.247195	−1.20156	−0.98509
2000	GRU + Attention	1.402577	2.512179	1.584985	−2.55559	−0.00118
2000	Transformer Encoder	0.719324	0.901978	0.949725	−0.27661	−0.16107
2000	FNN	5.946396	36.03272	6.002726	−49.9986	0.000607
2000	ResNet	0.918815	1.375626	1.172871	−0.94698	−0.81381
3000	BiLSTM	0.557207	0.6463	0.803928	0.235977	0.236167
3000	CNN + LSTM	0.998374	1.66297	1.289562	−0.96588	−0.00035
3000	CNN + GRU	2.25313	5.928203	2.43479	−6.00802	−0.00673
3000	CNN + Attention	1.482648	2.819197	1.679046	−2.33271	−0.07432
3000	CNN	1.565222	3.075744	1.75378	−2.63599	−0.06903
3000	VAE	0.263929	0.081713	0.285856	−592.812	−68.7838
3000	Simple RNN	0.809486	1.138417	1.066966	−0.34578	−0.2557
3000	GRU + Attention	0.749757	1.08692	1.042555	−0.2849	−0.00787
3000	Transformer Encoder	1.308901	2.337863	1.529007	−1.7637	−0.10741
3000	FNN	5.481222	30.79281	5.549127	−35.4017	−0.01322
3000	ResNet	1.065996	1.987914	1.409934	−1.35001	−1.25002
4000	BiLSTM	0.566931	0.584771	0.764703	0.158819	0.158916
4000	CNN + LSTM	0.779662	0.980019	0.989959	−0.40974	−0.00056
4000	CNN + GRU	1.251231	2.081645	1.442791	−1.9944	−0.01432
4000	CNN + Attention	0.851361	1.14326	1.069234	−0.64456	−0.0598
4000	CNN	0.799943	1.067414	1.033157	−0.53545	−0.07575
4000	VAE	0.187607	0.04504	0.212225	−733.128	−98.2667
4000	Simple RNN	0.805155	1.092853	1.045396	−0.57205	0.017145
4000	GRU + Attention	0.858142	1.138319	1.06692	−0.63745	0.001379
4000	Transformer Encoder	0.911492	1.340027	1.157595	−0.9276	−0.18881
4000	FNN	4.218891	18.40042	4.289572	−25.4686	−0.00789
4000	ResNet	1.04947	1.803121	1.342804	−1.59375	−1.08891
5000	BiLSTM	0.537969	0.503345	0.709468	0.26168	0.261796
5000	CNN + LSTM	0.681385	0.754899	0.868849	−0.10731	−0.00046
5000	CNN + GRU	0.642445	0.693075	0.832511	−0.01662	−0.00473
5000	CNN + Attention	0.747997	0.86742	0.931354	−0.27236	−0.03289
5000	CNN	0.666162	0.722275	0.849868	−0.05945	−0.02662
5000	VAE	0.12305	0.022333	0.149444	−509.388	−102.912
5000	Simple RNN	0.62642	0.663366	0.814473	0.026956	0.028359
5000	GRU + Attention	0.646964	0.705254	0.839794	−0.03449	−0.00028
5000	Transformer Encoder	0.770956	0.911904	0.954937	−0.33761	−0.07195
5000	FNN	1.085351	1.72848	1.314717	−1.53538	−0.08889
5000	ResNet	0.911236	1.315809	1.147087	−0.93007	−0.90149

a reliable choice for sequential data analysis. The GRU + Attention model also performs well by maintaining a balance between accuracy and computational efficiency, selectively focusing on important sequences to enhance predictions. On the other hand, CNN-based models, such as CNN + LSTM and CNN + GRU, exhibit significantly higher errors, particularly for smaller sample sizes, indicating their struggle in capturing long-range dependencies within the dataset. While CNN architectures are effective in feature extraction, their ability to model sequential relationships may be limited, leading to suboptimal performance. Similarly, ResNet, despite its deep learning capabilities, shows inconsistent results, often producing higher errors and poor R^2 scores, suggesting that residual learning techniques effective in image processing may not translate well to movie rating predictions. Meanwhile, Transformer Encoder and Simple RNN models perform moderately, though their higher variance in predictions suggests sensitivity to dataset size. Transformers generally require large amounts of data to perform optimally, while Simple RNNs are prone to vanishing gradient issues, making them less effective for long-term dependencies compared to GRU and LSTM-based models.

From a practical perspective, the trade-off between accuracy and generalization is crucial. While VAE provides the best accuracy in terms of error reduction, its poor R^2 and explained variance scores indicate that a model with slightly higher errors but better generalization, such as BiLSTM, may be preferable for real-world applications. Additionally, computational efficiency plays a vital role in model selection. Transformer-based architectures and deep models like ResNet, while powerful, are computationally expensive and may not be feasible in resource-constrained environments. In contrast, GRU + Attention offers a reasonable trade-off between accuracy and efficiency, making it a more practical choice. Furthermore, dataset size sensitivity is another

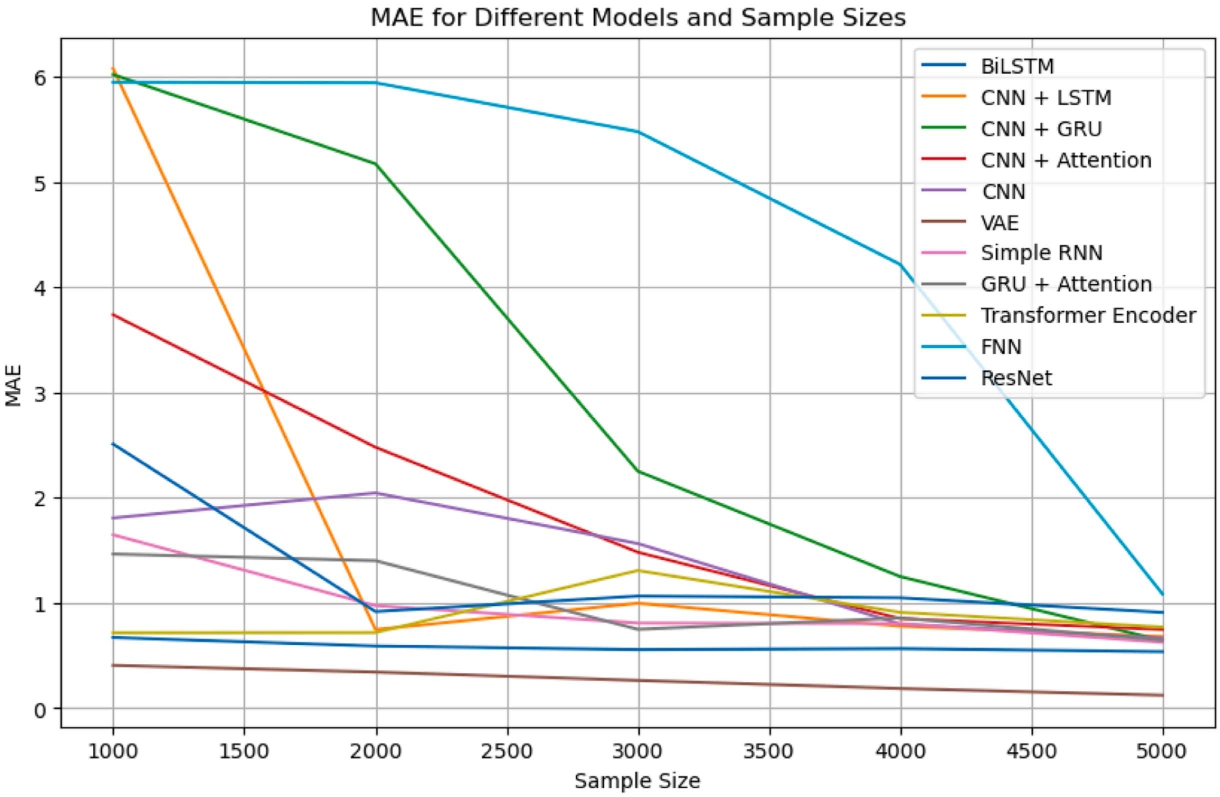


Fig. 2. MAE for Different Models and Different Sample Sizes.

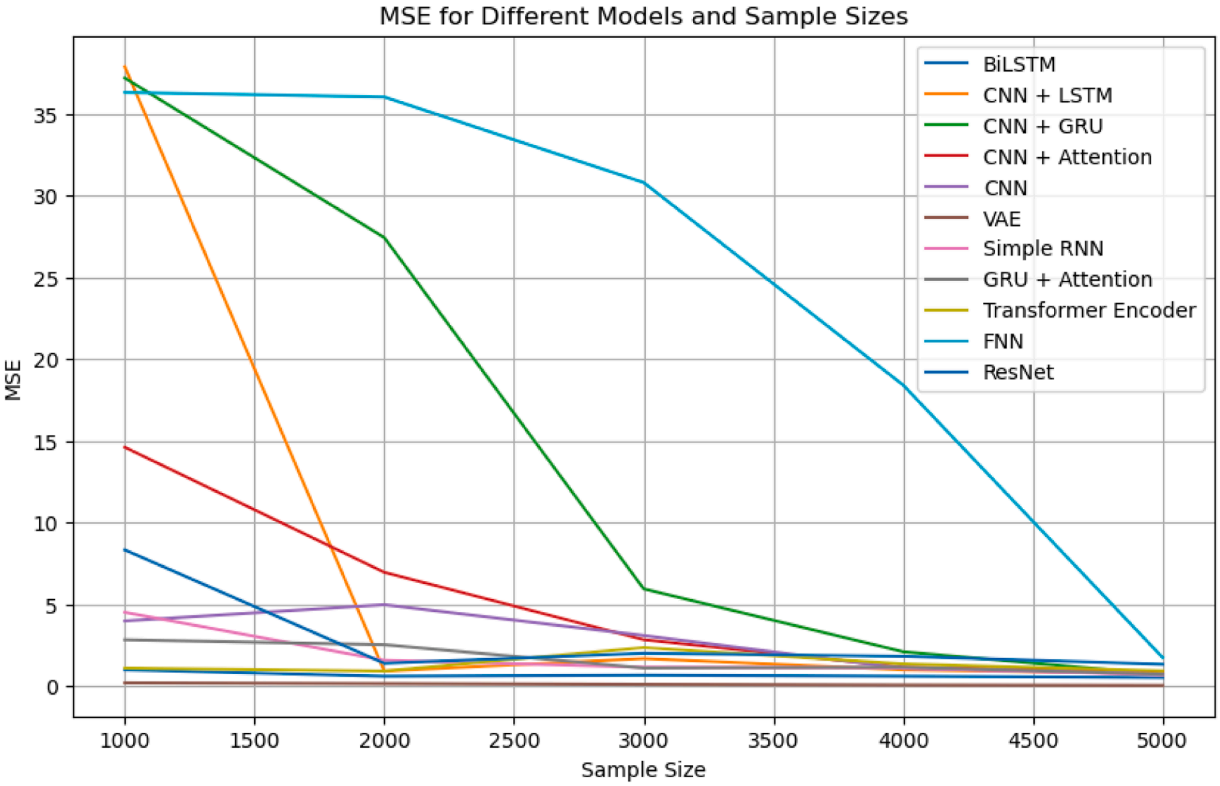


Fig. 3. MSE for Different Models and Different Sample Sizes.

critical factor, as models like CNN + LSTM and ResNet show performance degradation with smaller datasets, suggesting that they may require larger data volumes to fully leverage their architectural advantages. These findings underscore the importance of careful model selection based on practical constraints such as computational cost, dataset availability, and generalization ability, rather than relying solely

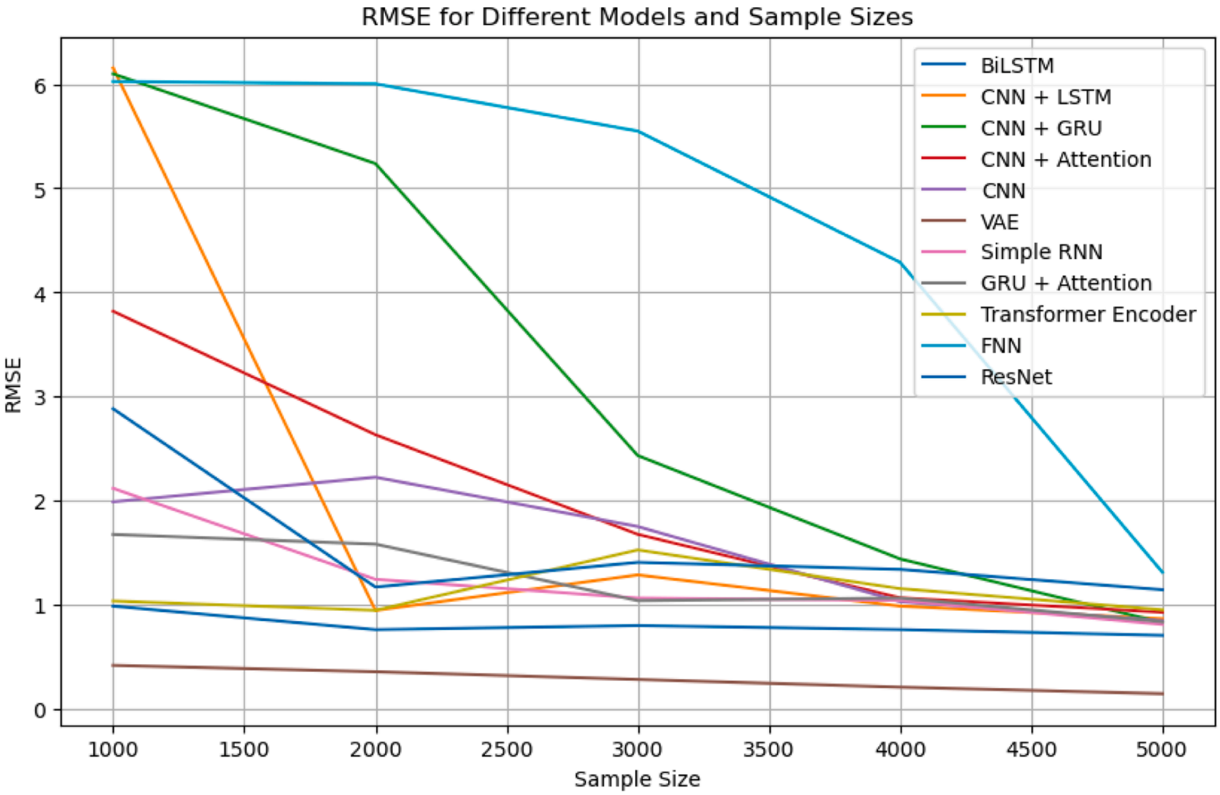


Fig. 4. RMSE for Different Models and Different Sample Sizes.

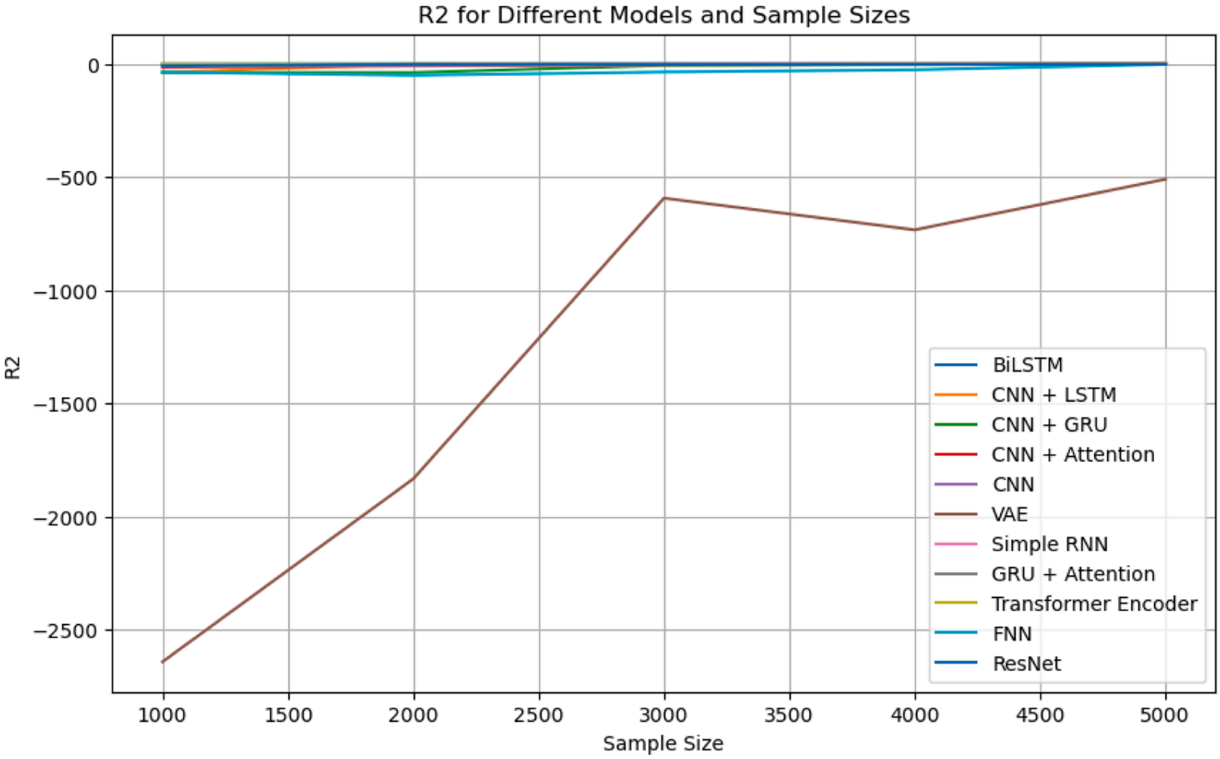


Fig. 5. R2 for Different Models and Different Sample Sizes.

on error metrics.

5.3. Model consistency

VAE demonstrates remarkably low MAE, MSE, and RMSE values across different sample sizes, indicating strong predictive performance in minimizing absolute errors. This suggests that VAE effectively captures latent patterns in the data, leading to precise individual predictions.

However, the model exhibits extremely negative R^2 values, indicating a significant discrepancy between the variance of the predicted and actual ratings. While this might initially suggest overfitting, it is more likely due to structural characteristics of the VAE rather than traditional overfitting in a deterministic model. Unlike standard regression-based models, VAE prioritizes reconstructing input features rather than directly optimizing for rating prediction, which may result in uninformative or misaligned feature representations.

Several factors could contribute to this behaviour:

- **Poor Latent Space Representation** – The learned latent space may not effectively capture the global variance of the target ratings, leading to inconsistent predictions.
- **Overly Strong KL Divergence Regularization** – Excessive regularization can force the latent space distribution too close to a prior (e.g., isotropic Gaussian), potentially limiting the expressiveness of the learned representations.
- **Mismatch Between Generative and Predictive Objectives** – VAE's primary goal is to generate meaningful representations of input data rather than directly minimize rating prediction error, which may cause it to underperform in tasks requiring strict numerical alignment.
- **Improper Feature Scaling or Suboptimal Hyperparameters** – Poorly scaled features, an inappropriate latent dimension size, or insufficient tuning of key hyperparameters may further degrade predictive performance.

To address these issues, future work could explore fine-tuning techniques such as adjusting the KL divergence weight, optimizing the latent space dimensionality, refining hyperparameters, and incorporating hybrid models that balance generative representation learning with explicit predictive objectives. These improvements could enhance VAE's interpretability while preserving its ability to capture complex feature interactions.

5.4. Practical implications and trade-offs

- Accuracy vs. Generalization: While VAE provides the best accuracy, its poor R^2 and explained variance scores highlight the importance of evaluating generalization. A model with slightly higher errors but better R^2, such as BiLSTM, may be preferable in real-world applications.
- Computational Cost vs. Performance: Transformer-based models and deep networks like ResNet are computationally expensive, making them impractical for resource-constrained environments. In contrast, GRU + Attention offers a reasonable trade-off between accuracy and efficiency.
- Dataset Size Sensitivity: Some models, such as CNN + LSTM and ResNet, perform worse on smaller datasets, indicating that they may require larger data volumes to leverage their architectural strengths effectively.

5.5. Impact of sample size

Increasing the sample size generally improves the performance of all the models, especially in context of reducing MAE and RMSE. However, from the Analysis of the Fig. 6, the improvement in R^2 and Explained Variance is not uniform, as some models still show adverse R^2, indicating poor fit despite the larger dataset.

5.6. Noteworthy performance

VAE consistently has low error metrics (MAE, MSE, RMSE), making it a candidate for further tuning, though the highly negative R^2 suggests it might require regularization or additional feature engineering to generalize better.

BiLSTM and GRU + Attention appears to be more balanced choices,

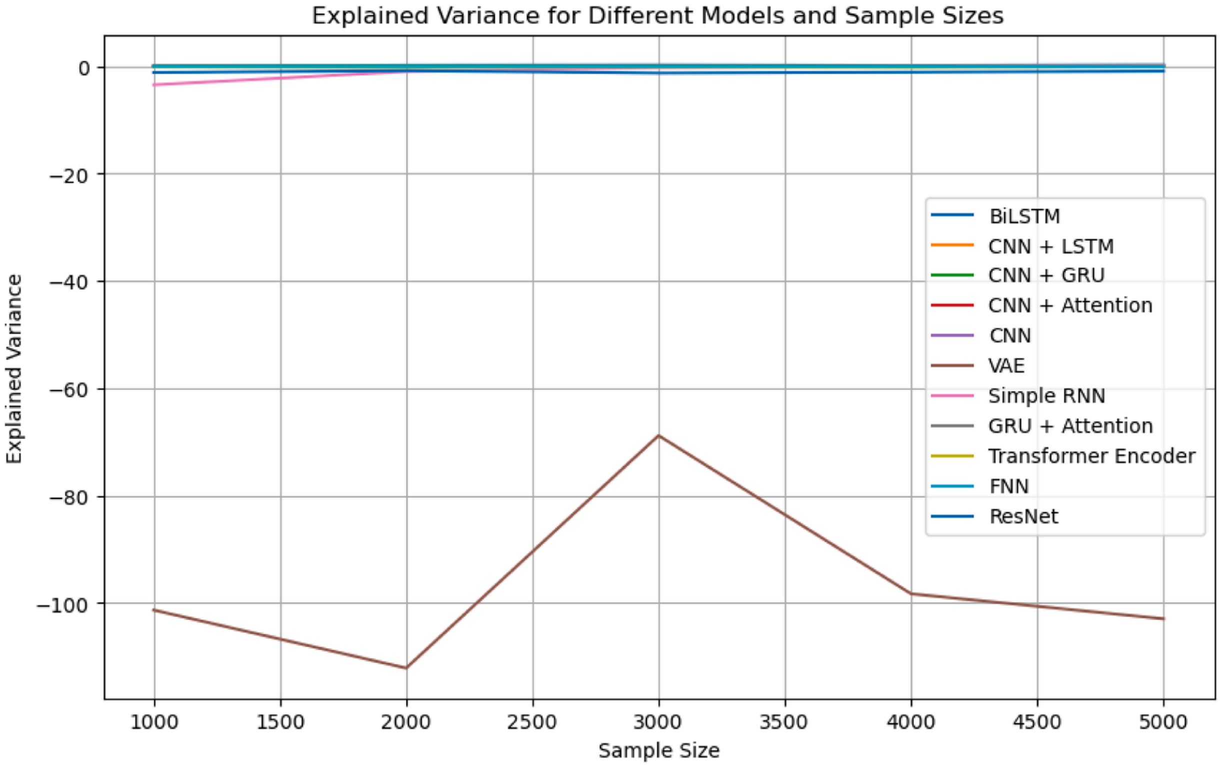


Fig. 6. Explained Variance for Different Models and Different Sample Sizes.

with moderate error metrics and reasonable R^2 values, representing both accuracy and generalizability.

Here are the context-specific definitions of the evaluation metrics, tailored for actual movie ratings (yi) predicted movie ratings (ŷi) :

Mean Absolute Error (MAE) measures the average magnitude of the errors which are in a set of predictions.

$$\mathbf{MAE} = (1/\mathbf{n}) * \Sigma|\mathbf{yi} - \hat{\mathbf{y}}\mathbf{i}| \tag{1}$$

Where, n: number of data points, yi: actual rating, ŷi: predicted rating.

Mean Squared Error (MSE) - measures the average squared difference, which is between the actual and predicted values.

$$\mathbf{MSE} = (1/\mathbf{n}) * \Sigma(\mathbf{yi} - \hat{\mathbf{y}}\mathbf{i})^2 \tag{2}$$

Root Mean Squared Error (RMSE) is the square root value, which is of the MSE, which is providing an error measure in the same units like as the original data.

$$\mathbf{RMSE} = \sqrt{\mathbf{MSE}} \tag{3}$$

R-squared (R^2) measures the proportion of the variance in the dependent variable (actual ratings) that is explained by the independent variable (predicted ratings).

$$\mathbf{R}^2 = 1 - \left(\frac{\mathbf{SSR}}{\mathbf{SST}}\right) \tag{4}$$

Where, SSR: Sum of Squared Residuals = Σ (yi - ŷi) ^{2}, SST: Total Sum of Squares = Σ(yi - ȳ)^2 , ȳ: mean of actual ratings

Explained Variance Score measures the proportion of variance in the dependent variable, explained the model predictions.

$$\mathbf{Explained\ Variance\ Score} = 1 - \left(\frac{\mathbf{Var}(\mathbf{y} - \hat{\mathbf{y}})}{\mathbf{Var}(\mathbf{y})}\right) \tag{5}$$

Where, Var (y - ŷ): variance of the residuals, Var(y): variance of the actual ratings

By calculating these kinds of metrics, we can compute the accuracy and reliability of our movie rating prediction model and make knowledgeable decisions about its performance and probable improvements.

In Fig. 7, The number of attention heads' sensitivity analysis in the CNN + Attention model shows that use of 4 attention heads results in the lowest loss (0.667), offering the best performance. Increasing the amount of attention heads beyond 4 leads to diminishing returns, with performance slightly degrading. Therefore, 4 heads strike the best balance between model complexity and performance.

The Summary of Differences between the given Models are:

VAE excels at handling uncertainty and missing data, and it is generative, meaning it is able to create new samples. This helps it overcome cold start problems more effectively than models like CNN + LSTM, CNN + GRU, or BiLSTM, which are not designed for generative tasks.

CNN-based models (LSTM, GRU, Attention) are intended for sequential data processing (such as text or time-series) and capture temporal dependencies. However, they still struggle with cold-start problems as they do not generate new data which relies heavily on the availability of historical data.

BiLSTM and CNN + Attention are mainly decent at capturing complex sequential dependencies, but they are not inherently generative as VAE. Their attention mechanisms help the model focus on important sequences or features, but they still require explicit handling of missing data and cold-start issues.

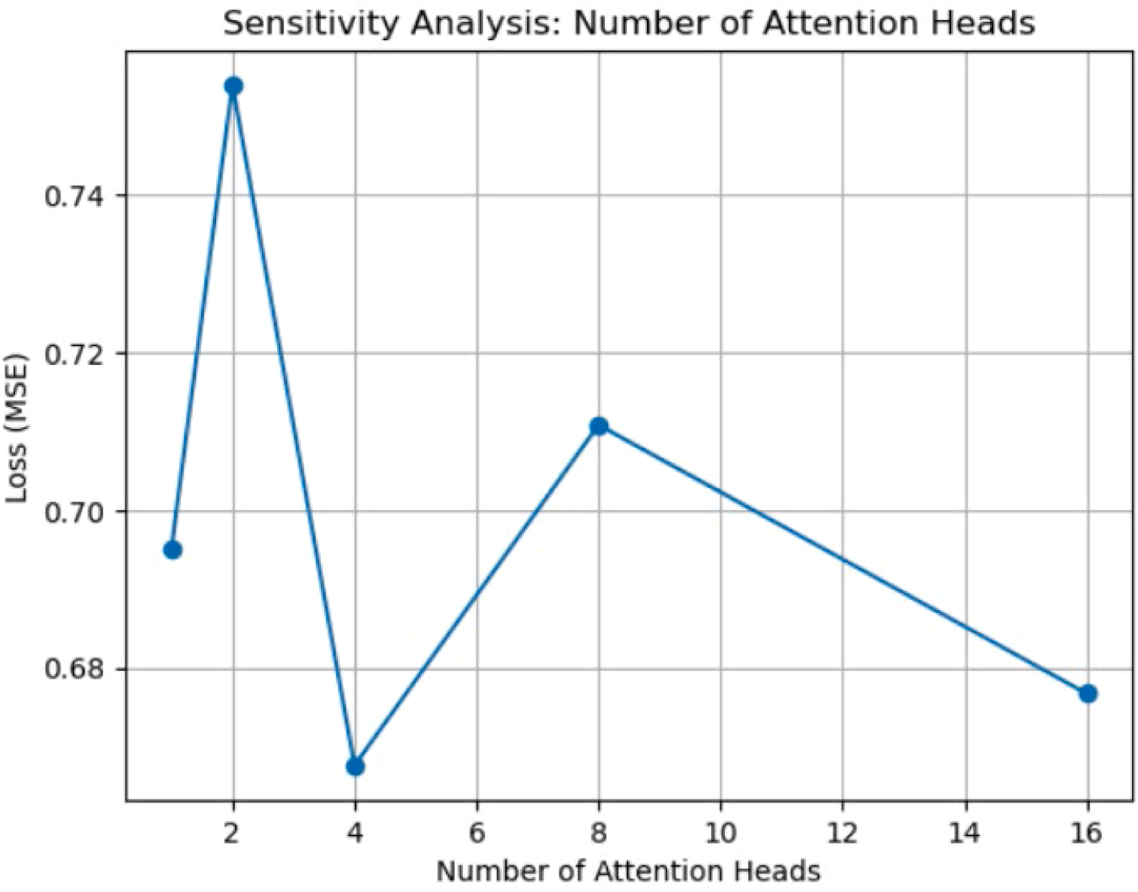


Fig. 7. Sensitivity Analysis:Number of Attention Heads.

Training Time fluctuates significantly. VAE tends to be the slowest due to its complex training process which involves variational inference. The CNN-based models and BiLSTM can have moderate training times.

CNN + LSTM efficiently extracts features and captures temporal patterns, but struggles with long sequences. CNN + GRU offers better efficiency but may miss long-range dependencies. CNN + Attention improves performance with a focus mechanism, though it adds complexity. VAE excels at learning a regularized latent space but can struggle with noisy data. Simple RNNs are efficient but fail with long-term dependencies. GRU + Attention combines efficiency with attention but still faces long-range challenges. FNN is simple but lacks the ability to model complex relationships, while ResNet helps with gradient flow but can lead to overfitting. Transformer models capture long-range dependencies well but are computationally expensive, and GANs are powerful but often unstable during training.

The VAE model performs significantly better than SVR on the basis of provided MAE and MSE. The VAE's MAE decreases from 0.4066 to 0.1231, and MSE drops from 0.1775 to 0.0223, indicating improved accuracy with each iteration. In contrast, as shown in [34], the SVR model has higher MAE (0.787) and MSE (1.097), demonstrating that VAE minimizes prediction errors more effectively. Table 3.

This extensive evaluation and comparison framework provides the most effective Deep Learning Model for Movie rating Predictions.

5.7. Comparison with other benchmark dataset

To ensure a thorough comparison, we evaluate our proposed approach against the widely used MovieLens [35] benchmark. The MovieLens-based method chiefly relies on structured numerical and categorical features such as age, gender, occupation, and movie year, with TF-IDF vectorization applied to movie titles, followed by dimensionality reduction using Truncated SVD and feature scaling. In contrast, proposed customized database-based method integrates both structured and unstructured data, incorporating user reviews, sentiment scores extracted using DistilBERT, and user ratings. By leveraging TF-IDF vectorization with a higher dimensionality (10,000 features) and applying Truncated SVD (200 components), our method captures richer contextual information from textual data. Unlike MovieLens, which focuses on predefined user and item attributes, our approach enhances predictive performance by incorporating sentiment polarity and user-generated content, making it more effective in capturing nuanced user preferences.

Table 4
The evaluation metrics for given different models on movielens dataset.

Model	MSE	MAE	RMSE
BiLSTM	1.263141	0.943774	1.123895
CNN + LSTM	1.263239	0.941157	1.123939
CNN + GRU	1.263117	0.941631	1.123885
CNN + Attention	1.263012	0.942678	1.123838
CNN	1.263223	0.941211	1.123932
Simple RNN	1.263106	0.941687	1.123880
GRU + Attention	1.263023	0.942980	1.123843
FNN	1.265930	0.947997	1.125135
ResNet	1.263187	0.941339	1.123916
VAE	0.084038	0.250673	0.289893
Transformer	0.080232	0.242910	0.283253

Table 3
Comparing Models with respect to features.

Feature	VAE	CNN + LSTM	CNN + GRU	CNN + Attention	BiLSTM
Model Type	Probabilistic Deep Learning Model	Convolutional + Recurrent Model	Convolutional + Recurrent Model	Convolutional + Attention-based Model	Recurrent Deep Learning Model
Latent Space Representation	Latent probabilistic space	Does not have explicit latent space	Does not have explicit latent space	Attention mechanism instead of latent space	Sequential hidden states
Handling Uncertainty	Models' uncertainty using latent space	Does not model uncertainty	Does not model uncertainty	Attention weights can focus on key areas, but no explicit uncertainty modeling	Does not model uncertainty
Generative Aspect	Generates new data (user-item interaction)	Not generative, focuses on prediction	Not generative, focuses on prediction	Focused on learning attention-based relationships	Not generative, focuses on prediction
Regularization	KL divergence regularization for smoothness	Regularization through dropout and L2	Regularization through dropout and L2	Regularization via attention and dropout	Regularization via L2 and dropout
Handling Missing Data	Handles missing data via latent space representation	Requires imputation or missing data strategy	Requires imputation or missing data strategy	Requires imputation or missing data strategy	Requires imputation or missing data strategy
Data Type Handling	Can handle complex data distributions due to probabilistic nature	Focuses on sequential data (e.g., time series, text)	Focuses on sequential data (e.g., time series, text)	Focuses on sequential data with attention mechanism	Focuses on sequential data (text or time series)
Feature Interactions	Models complex interactions in latent space	Captures spatial and temporal interactions	Captures spatial and temporal interactions	Focuses on key features using attention weights	Models sequential interactions
Scalability	Can scale but may be slow due to sampling in training	Scales well but requires sufficient computational resources	Scales well but requires sufficient computational resources	Scales well with attention mechanism, but needs careful tuning	Scales well for sequential data
Overfitting Prevention	KL Divergence term helps to prevent overfitting	Dropout layers for regularization	Dropout layers for regularization	Dropout + attention regularization	Dropout regularization
Training Complexity	High computational complexity due to sampling from latent space	Requires tuning of both CNN and LSTM parameters	Requires tuning of both CNN and GRU parameters	Requires tuning of CNN + Attention weights	Requires tuning of LSTM parameters
Flexibility	High flexibility due to the generative model	Flexible for sequence-based problems	Flexible for sequence-based problems	Flexible for sequence-based problems with attention focus	Flexible for sequence-based problems
Cold Start Problem	Handles cold start better through generative nature	Struggles with cold start if no historical data	Struggles with cold start if no historical data	Struggles with cold start if no historical data	Struggles with cold start if no historical data
Hyperparameter Tuning	Needs tuning of latent space size, learning rate, and regularization terms	Needs tuning of CNN layers, LSTM parameters	Needs tuning of CNN layers, GRU parameters	Needs tuning of CNN layers, attention parameters	Needs tuning of LSTM parameters
Interpretability	Lower interpretability due to the complex latent space and probabilistic nature	Moderate interpretability in terms of learned filters and sequential patterns	Moderate interpretability in terms of learned filters and sequential patterns	Lower interpretability due to attention mechanisms being black-box	Moderate interpretability in terms of sequential patterns
Training Time	Can be slow due to variational inference and sampling steps	Moderate training time due to sequential data processing	Moderate training time due to sequential data processing	Moderate to high depending on the attention complexity	Moderate to high depending on data size

Here is the formatted Table 4 with all the models and their evaluation metrics:

The VAE and Transformer models have significantly lower errors than the others, indicating superior performance.

Although customized database-based method requires higher computational resources due to transformer-based sentiment analysis, it provides a more comprehensive understanding of user sentiment and engagement, demonstrating its advantage in real-world movie recommendation scenarios where textual opinions significantly influence user decisions.

The evaluation dataset is designed to ensure diversity by incorporating a broad range of movies across multiple genres, different user demographics, and varying sentiment expressions in reviews. Unlike traditional datasets that primarily rely on structured numerical features (e.g., MovieLens), proposed dataset integrates textual data from IMDb user reviews, enriched with sentiment scores extracted using DistilBERT. This approach enables a more nuanced analysis of user preferences beyond explicit ratings. Additionally, the dataset includes movies from different years and a variety of user profiles, ensuring that the proposed method is robust across different audience segments and rating behaviors. By leveraging both structured and unstructured data, our evaluation framework effectively highlights the strengths of different models in handling diverse user interactions and contextual factors in movie recommendation.

The dataset is randomly sampled from a large corpus to ensure diversity across different attributes like genres, directors, actors, and sentiments. Additionally, DistilBERT-based sentiment extraction captures nuanced variations, and TF-IDF with SVD retains key textual diversity. Expanding the sample size or incorporating stratified sampling can further enhance representativeness.

MovieLens 100 K Dataset (structured format with user-item interactions) Fields are: user_id, item_id, rating, timestamp, movie_title, year, age, gender, occupation, zip_code. Models like BiLSTM, CNN + LSTM, CNN + GRU, CNN + Attention, etc., were evaluated on MSE, MAE, and RMSE. VAE and Transformer models performed significantly better.

The Proposed Dataset with Movie Metadata & Sentiment Analysis Fields are director_name, actor_1_name, actor_2_name, actor_3_name, genres, movie_title, comb, User_Score, Review, Review_Sentiment Performance was measured across different sample sizes (1000 to 5000) for multiple models. VAE again showed the best performance, with Transformer Encoder also performing well.

Both MovieLens 100 K and our dataset support movie rating prediction, but MovieLens 100 K focuses on structured user-item interactions, while our dataset integrates metadata and sentiment analysis. Unlike MovieLens, our dataset leverages textual and semantic features, improving model performance, especially for VAE and Transformer-based models. This broader feature set provides a richer benchmark, capturing deeper user preferences beyond explicit ratings.

- **Similarities:** Both MovieLens 100 K and our dataset serve as benchmarks for movie rating prediction and support various deep learning models.
- **Differences:** MovieLens 100 K is structured around user-item interactions, including demographic information, whereas our dataset incorporates additional metadata (director, actors, genres) and sentiment analysis from reviews, providing richer contextual information.
- **Advantages:** The inclusion of sentiment-based and metadata-driven features enhances predictive performance, particularly for complex models like VAE and Transformer Encoder. This broader feature representation enables a more nuanced understanding of user preferences beyond explicit numerical ratings, making our dataset a more comprehensive benchmark.

6. Conclusion

This study delivers a thorough performance analysis of numerous deep learning models for movie rating prediction, by examining architectures such as BiLSTM, CNN + GRU, CNN + LSTM, CNN + Attention, VAE, and other advanced frameworks. Through evaluating these kinds of models across manifold metrics, including MAE, MSE, RMSE, R^2, and Explained Variance, clear patterns in model performance are identified and effectiveness is found for accurate rating prediction. The results show that while VAE steadily attains the highest accuracy, attention-based models offer valuable improvements in interpretability as well as adaptability to varying input sequences. Models like CNN and BiLSTM also demonstrate reliable performance, and they are also balancing accuracy with computational efficiency. These types of findings underscore the importance of picking the accurate architecture based on the specific requirements of recommendation systems, whether prioritizing prediction accuracy, interpretability, or computational efficiency. This study very well contributes a benchmark for deep learning models in movie rating prediction, which is guiding researchers and practitioners toward optimized model selection in personalized recommendation contexts. Based on the evaluation metrics, the VAE model constantly outperforms others across all sample sizes with the lowermost MAE (0.123 for 5000 samples), MSE (0.022), and RMSE (0.149), which is demonstrating its superior predictive accuracy. However, its negative R^2 and Explained Variance still suggest potential limitations in capturing data variability, which is warranting further exploration. The proposed approach integrating sentiment analysis improves movie rating prediction accuracy compared to traditional methods and outperforms benchmark datasets like MovieLens in capturing user preferences. Future research may discover integrating these kinds of various models or incorporating hybrid architectures to further improve the evaluation measure like prediction accuracy and model robustness. The paper could benefit from outlining specific improvements for hybrid models, by integrating reinforcement learning for adaptive recommendations and addressing data sparsity issues. Exploring hybrid models with advanced optimization techniques, such as Bayesian optimization, could enhance accuracy. Additionally, incorporating real-world factors like user behavior patterns and explainable AI techniques can make the system more practical and interpretable.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work the author(s) used ChatGPT in order to improve grammar. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

CRediT authorship contribution statement

Manisha Valera: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Resources, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Dr. Rahul Mehta:** Writing – review & editing, Validation, Supervision, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data Available Upon Request.

References

[1] H. Xia, J.J. Li, Y. Liu, Collaborative filtering recommendation algorithm based on attention GRU and adversarial learning, IEEE Access. 8 (2020) 208149–208157, https://doi.org/10.1109/ACCESS.2020.3038770.

[2] Xia L., Yang Y., Chen Z., Yang Z., Zhu S. Movie recommendation with poster attention via multi-modal transformer feature fusion. arXiv preprint arXiv:2407.09157. (2024) Jul 12.

[3] Bahare Askari, Jaroslaw Szlichta, Amirali Salehi-Abari, Variational autoencoders for top-k recommendation with implicit feedback, in: Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval, 2021.

[4] Shiri, Farhad Mortezapour, Thinagaran Perumal, Norwati Mustapha, and Raihani Mohamed, A comprehensive overview and comparative analysis on deep learning models: CNN, RNN, LSTM, GRU, arXiv preprint arXiv:2305.17473, (2023).

[5] H. Wang, N. Lou, Z. Chao, A personalized movie recommendation system based on LSTM-CNN, in: 2nd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI), 2020 (2020).

[6] D. Liang, R. Krishnan, T. Jebara, Variational autoencoders for collaborative filtering, in: Proceedings of the 2018 World Wide Web Conference (WWW '18). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, 2018, pp. 689–698. CHE.

[7] Sophort Siet, Sony Peng, Sadriddinov Ilkhomjon, Misun Kang, Doo-Soon Park, Enhancing sequence movie recommendation system using deep learning and kmeans, Appl. Sci. 14 (6) (2024) 2505.

[8] C. Dang, M. Moreno-García, F. De la Prieta, An approach to integrating sentiment analysis into recommender systems, Sensors 21 (16) (2021) 5666.

[9] Chandrakala Arya, Enhancing movie recommendation systems with deep learning and sentiment analysis, Int. J. Mech. Eng. 6 (3) (2021). ISSN: 0974-5823.

[10] H. Xia, Y. Luo, Y. Liu, Attention neural collaboration filtering based on GRU for recommender systems, Complex. Intell. Systems. (2021) 1367–1379.

[11] J. Panchal, S. Vanjale, A deep learning approach towards cold start problem in movie recommendation system, Int. J. Recent Innov. Trends Comput. Commun. 11 (8) (2023). VolumeIssueISSN: 2321-8169.

[12] D. Alsaleh, S. Larabi-Marie-Sainte, Arabic text classification using convolutional neural network and genetic algorithms, IEEE Access. 9 (2021) 91670–91685.

[13] J. Kufel, et al., What is machine learning, artificial neural networks and deep learning?—Examples of practical applications in medicine, Diagnostics 13 (15) (2023) 2582.

[14] J. Heaton, Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep learning, Genet. Program. Evolvable Mach. 19 (1–2) (2018) 305–307.

[15] S.V. Georgakopoulos, S.K. Tasoulis, A.G. Vrahatis, V.P. Plagianakos, Convolutional neural networks for toxic comment classification, in: Proceedings of the 10th Hellenic conference on artificial intelligence, 2018, pp. 1–6.

[16] N.I. Widiastuti, "Convolution Neural Network for Text Mining and natural language Processing, IOP Conference Series: materials Science and Engineering, (2019), no. 5, p. 052010.

[17] O.A. Montesinos López, A.M. López, and J. Crossa, Convolutional neural networks, multivariate statistical machine learning methods for genomic prediction, (2022) , pp. 533–577.

[18] I. Dhall, S. Vashisth, G. Aggarwal, Automated hand gesture recognition using a deep convolutional neural network model, in: 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence), 2020, pp. 811–816.

[19] E.C. Nisa, Y.Der Kuan, Comparative assessment to predict and forecast water-cooled chiller power consumption using machine learning and deep learning algorithms, Sustain. (Switzerland) 13 (2) (2021) 1–18.

[20] M. Azizjon, A. Jumabek, W. Kim, 1D CNN based network intrusion detection with normalization on imbalanced data, in: International Conference on Artificial Intelligence in Information and Communication, ICAIIC, 2020, pp. 218–224.

[21] W. Ramadhanti, E.B. Setiawan, Topic detection on twitter using deep learning method with feature expansion GloVe, Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI) 9 (3) (2023) 780–792.

[22] Arliyanna Nilla, Erwin Budi Setiawan, Film recommendation system using content-based filtering and the convolutional neural network (CNN) classification methods, Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI) (2024) 17–29.

[23] V. Sanh, L. Debut, J. Chaumond, T. Wolf, DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter, arXiv preprint (2020). arXiv:1910.01108v4 [cs.CL].

[24] S. Abdel-Salam and A. Rafea, Performance study on extractive text summarization using BERT models, information, (2022), 13(2), 67.

[25] A. Alshanqiti, A. Namoun, A. Alsughayir, A. Almashreqi, A. Gilal, S. Albouq, Leveraging DistilBERT for summarizing arabic text: an extractive dual-stage approach, IEEE Access. 10 (2021) 312–325.

[26] Zakir Mujeeb Shaikh, Suguna Ramadass, Unveiling deep learning powers: LSTM, BiLSTM, GRU, BiGRU, RNN comparison, Indones. J. Electr. Eng. Comput. Sci. 35 (1) (2024) 263–273. ISSN: 2502-4752.

[27] Haitao He, Zhifu Shang, Mingjie Wu, Yuling Zhang, Movie recommendation system based on traditional recommendation algorithm and CNN model, Highl. Sci. Eng. Technol. 34 (2023). Volume.

[28] TMDB Website, Available: https://www.themoviedb.org/.

[29] imdb-5000-movie-dataset, Available: https://www.kaggle.com/datasets/carolzhangdc/imdb-5000-movie-dataset.

[30] The Movies Dataset, Available: https://www.kaggle.com/datasets/rounakbanik/the-movies-dataset.

[31] List of American films of 2018, Available: https://en.wikipedia.org/wiki/List_of_American_films_of_2018.

[32] List of American films of 2019, Available: https://en.wikipedia.org/wiki/List_of_American_films_of_2019.

[33] List of American films of 2020, Available: https://en.wikipedia.org/wiki/List_of_American_films_of_2020.

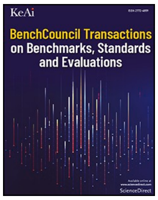
[34] Manisha Valera, Rahul Mehta, Comprehensive assessment and optimization of sentiment analysis models for movie reviews with enhanced movie recommendation systems, SSRG int. J. Electron. Commun. Eng. 11 (12) (2024) 258–271, https://doi.org/10.14445/23488549/IJECE-V11I12P124. Crossref.

[35] https://grouplens.org/datasets/movielens/100k/.

Contents lists available at ScienceDirect

BenchCouncil Transactions on Benchmarks, Standards and Evaluations

journal homepage: www.keaipublishing.com/en/journals/benchcouncil-transactions-onbenchmarks-standards-and-evaluations/



Full Length Article

AI-powered Mathematical Sentiment Model and graph theory for social media trends

M. VENKATACHALAM [a], R.VIKRAMA PRASAD [b,*]

[a] Research Scholar, Department of Mathematics, Government Arts College (Autonomous), Salem-7, Tamilnadu, India
[b] Assistant Professor, Department of Mathematics, Government Arts College (Autonomous), Salem-7, Tamilnadu, India

ARTICLE INFO

Keywords:
Graph theory
Markov decision process
Sentiment analysis
Monkeypox
Knowledge graph

ABSTRACT

Significant issues have arisen as a result of the global spread of monkeypox, such as the extensive transmission of false information, public fear, and stigmatization on social media. Increased fear, prejudice, stigmatization of minority groups, and opposition to public health initiatives are frequently the results of these problems. Furthermore, health authorities are unable to provide correct information and prompt actions due to a lack of efficient methods for analyzing the enormous amounts of unstructured social media data. This disparity weakens crisis management initiatives and increases public skepticism of health guidelines. In order to address these issues, this study looks into the attitude around monkeypox on social media in order to pinpoint public worries, counter false information, and enhance communication tactics. The study intends to improve public comprehension, offer practical insights, and help health authorities manage the outbreak by fusing graph theory with AI-driven sentiment analysis. In order to facilitate semantic analysis of tweets through structured information extraction, graph theory is used to organize unstructured or semi-structured data by creating meaningful links between entities. Furthermore, opinions on monkeypox infection in social media are analyzed and user sentiments are detected using a reinforcement Markov decision process. According to experimental results, the suggested model's accuracy on the Monkeypox tweet dataset was 98 %. These results help raise awareness of monkeypox among the general population and promote an educated and robust social response.

1. Introduction

In 1958, the monkeypox virus was initially identified in research-breed monkey colonies. In 1970, the first recorded human case of monkeypox virus occurred in the Democratic Republic of the Congo. Vaccines against the virus have since been created [1]. The monkeypox virus was deemed exterminated in 1980, and population immunization was discontinued. The fatality rate during a monkeypox outbreak has historically ranged from 1 % to 10 %, despite the fact that the majority of patients may recover. Originally affecting African nations, monkeypox is an infectious illness that has recently spread to almost every city on the planet. Although the world health organization does not recognize it as a pandemic, some experts believe it should be treated as such [2]. Many articles and comments on the symptoms, treatments, side effects, and other people's thoughts about the monkeypox virus have been made on social media sites like Reddit and Twitter. To find patterns and trends, it's critical to examine these user-generated materials [3]. The same tactics may be applied in the event of monkeypox. There are very limited early studies reported for understanding the general public's attitude toward monkeypox or general analysis, but a detailed analysis should be carried out to get a clear picture of the trends and facts [4]. Given the recent spread of monkeypox, associated digital information and opinions have also spread on different social media platforms, including Twitter. Determining the public trends and views about monkeypox is fundamental for governments, policymakers, healthcare providers, and researchers to use the available resources to control and mitigate the burden of the recent outbreak in an efficient and timely manner [5]. Opinion mining primarily deals with a person's concrete view of something, while sentiment refers to an attitude or thought prompted by a feeling [6]. Sentiment analysis and opinion mining [7] were initially used for product review applications but have recently shifted to other tasks, including: stock markets, elections, disasters, healthcare and software engineering [8]. The content shared across social media platforms provides a valuable source of knowledge about the physical environment and social phenomena [9]. As a result, the public security domain has become an important application domain in sentiment

[*] Corresponding author.
E-mail address: vikramprasad20@gmail.com (R. PRASAD).

https://doi.org/10.1016/j.tbench.2025.100202
Received 9 December 2024; Received in revised form 18 March 2025; Accepted 21 April 2025
Available online 17 May 2025

analysis and opinion mining [10].

In the sentiment analysis and opinion mining, graph theory plays a pivotal role by enabling the modeling of entities and their connections as networks [11]. In weighted graphs, edges carry sentiment scores enable the quantification and sentiment flow across the network, offering deeper insights into the dynamics of public opinion [12]. Knowledge graphs [13], a specialized application of graph theory, extend this concept by integrating semantic relationships between entities. Techniques such as spectral clustering, graph neural network (GNN) [14] and diffusion modeling can predict how sentiments evolve over time or simulate the impact of interventions, such as targeted awareness campaigns. By capturing both structural and contextual relationships, graph theory not only enhances the understanding of public sentiment but equips policymakers with actionable insights to address public concerns effectively during outbreaks like monkeypox [15]. Table 1 shows the detailed analysis of sentiment analysis uses AI, which provides creative ways to examine enormous volumes of unstructured social media data.

Table 1
Existing state-of-art works on sentimental analysis on healthcare social media trends.

Ref.	Method	Key features	Contributions	Performance
[16]	MWGCN	Local Context Weighted Graph (LCG), Multigrain Dot-Product Weighting (MGDW)	Reduces long-distance dependencies, emphasizes local context	Improves sentiment classification accuracy
[17]	GCN	Graph structure-based learning	Captures contextual relationships in data	Accuracy increased by 78 % over baseline
[18]	Combination Graph with KL-divergence	KL-divergence between likelihood models	Enhances in formativeness of nodes in graph-based models	Better structured learning performance
[19]	LSTM + *N*-gram Graph-Cut	Sequence modeling with LSTM and N-gram graph structure	Improves feature extraction and representation	9 % accuracy gain in three-way classification
[20]	Semantic-HGCN	Hierarchical semantic graph encoding	Models multi-level semantic relationships	Improved sentiment prediction
[21]	FGCN	Fuzzy logic integrated with GCN layers	Reduces ambiguity in sentiment detection	Enhances robustness of sentiment classification
[21]	BERT + BiLSTM + GCN	Contextual embedding with BERT-BiLSTM and fuzzy adjacency	Captures deep semantic and structural features	Improves interpretability and classification performance
[22]	Graph using Publication Attributes	Uses attributes like authors, keywords to build graphs	Organizes topics with Louvain community detection	Better thematic structure in sentiment analysis
[23]	Hierarchical Graph Contrastive Learning	Learns local and global representations	Captures complex relationships in utterances	Enhances multimodal sentiment extraction
[24]	MGMFN (GNN + MLP-Mixer)	Combines multiple GNN graphs and long-range MLP features	Strengthens spatial and semantic representation	83.72 % and 86.43 % accuracy in Chinese text classification
[25]	GNN + RF	Uses GNN for learning and Random Forest for classification	Analyzes user attitudes from social data (ChatGPT tweets)	Efficient multi-class sentiment categorization

From the review [11–26], we found the problems associated with using AI-powered sentiment analysis and graph theory for analyzing monkeypox social media trends. While techniques like multi-weight graph convolutional network (MWGCN) and fuzzy graph convolutional networks (FGCN) address local context and syntactic features, their adaptation to monkeypox-specific discussions remains limited [16]. Handling unstructured social media data is a persistent challenge due to its noisy and multimodal nature. Although graph-based methods like Semantic-HGCN and fuzzy logic [17,18]integration help structure data, their scalability for large-scale real-time analysis is insufficient. Existing models, such as hierarchical graph contrastive learning, excel in multimodal sentiment extraction but fail to explicitly tackle the dynamics of misinformation and its impact. Moreover, current sentiment analysis approaches often lack the capacity to analyze interconnected factors such as stigma, misinformation, and public health narratives within a single framework. While hybrid methods like N-gram Graph-Cut [18] combined with LSTM improve accuracy, their application to evolving social media trends and high-dimensional datasets faces scalability challenges. The inability to adequately represent complex social relationships [16–25] is another problem, as discussions about monkeypox often involve intricate interactions between users, groups, and topics. Methods like adjacency graphs and the Louvain algorithm provide structural insights but are limited in capturing nuanced interdependencies in this context [19]. Bias and stigmatization against minority groups also complicate sentiment analysis, requiring models to account for these ethical concerns while ensuring interpretability [21, 22]. Despite achieving high accuracy, such as 98 % in monkeypox sentiment classification, AI models lack transparency, hindering their adoption by public health authorities. The limited integration of domain-specific health knowledge and the challenges in adapting to real-time trends restrict the effectiveness of current approaches, making it crucial to develop models that are both adaptable and context-aware.

This study introduces an innovative approach that leverages AI-powered sentiment analysis and graph theory to gain valuable insights into social media trends surrounding monkeypox. By integrating these advanced techniques, the work aims to enhance public understanding, provide actionable insights, and support health authorities in effectively managing the outbreak. The primary contributions of this research are summarized as follows:

1. To address the challenges of analyzing unstructured or semi-structured social media data, graph theory is employed to establish meaningful connections between various entities such as keywords, hash-tags, and user interactions. This facilitates semantic analysis of tweets, transforming chaotic data into an organized framework that can be effectively analyzed for sentiment and thematic trends.
2. The study utilizes a reinforcement Markov decision process (RMDP) to analyze opinions and detect user sentiments regarding monkeypox-related discussions on social media. It enables a nuanced understanding of public perception, including the identification of misinformation, stigmatization, and emotional responses, which are critical for devising targeted communication strategies.
3. The proposed methodology is validated using a comprehensive Monkeypox tweet dataset comprising 61,379 tweets collected from Twitter between May 7 and June 11, 2022. The results showed the model's high accuracy in sentiment analysis and potential to uncover meaningful insights, contributing to resilient public response to the monkeypox outbreak.

2. Material and methods

Fig. 1 illustrates the workflow of the AI-driven sentiment analysis model designed for detecting monkeypox infection sentiment using game theory. The process begins with the Monkeypox tweet dataset, comprising 61,379 tweets published on Twitter between May 7 and June 11, 2022.

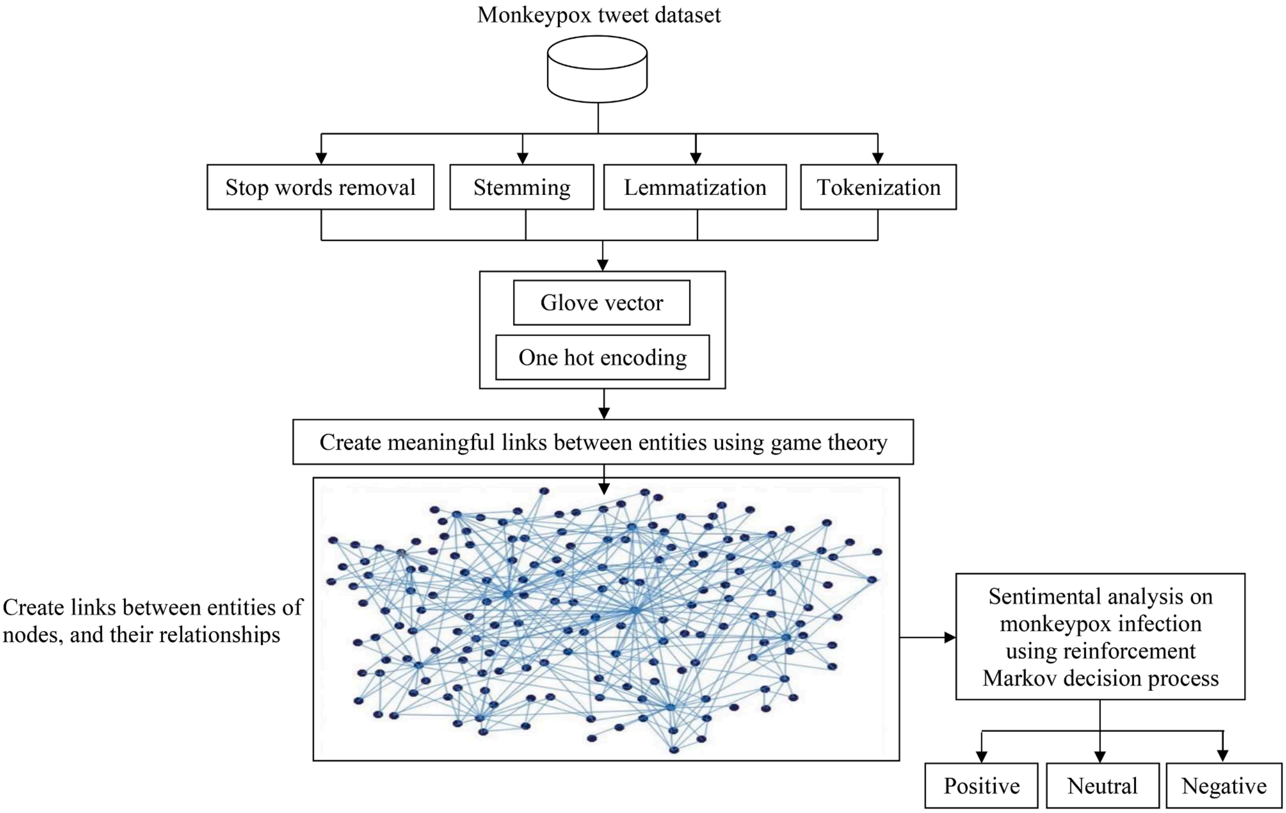

Monkeypox tweet dataset
Stop words removal
Stemming
Lemmatization
Tokenization
Glove vector
One hot encoding
Create meaningful links between entities using game theory
Create links between entities of nodes, and their relationships
Sentimental analysis on monkeypox infection using reinforcement Markov decision process
Positive
Neutral
Negative

Fig. 1. AI-driven sentiment analysis for monkeypox infection detection using game theory.

The raw textual data undergoes preprocessing steps to ensure its readiness for analysis. Tokenization then splits the text into smaller units, such as words or phrases, enabling efficient processing. After preprocessing, feature extraction techniques are applied to represent the textual data numerically. GloVe vector representation is used to embed words into high-dimensional vector spaces, capturing their semantic and contextual relationships. Additionally, one-hot encoding convert's text into binary vectors, ensuring each unique word is represented distinctly. Next, game theory is employed to create a graph that establishes meaningful links between entities. Nodes in the graph represent entities like keywords, hash tags, user mentions, or topics, while edges are formed based on semantic or contextual similarities. Game-theoretic principles evaluate the importance of these relationships, ensuring that the graph highlights significant patterns and connections among the data. The sentiment classification is performed using a reinforcement Markov decision process (RMDP), which categorizes tweets into positive, neutral, or negative sentiments. RMDP operates by iteratively learning optimal policies through a state-action framework, where states represent the current sentiment context and actions assign appropriate sentiment labels based on extracted features. This iterative learning ensures precise classification by continuously refining predictions. By integrating graph theory for semantic analysis and leveraging RMDP for classification, the model effectively captures public sentiment trends on monkeypox.

2.1. *Data source and description*

The open-source website GitHub provided the Monkeypox Twitter dataset used in this study, which consists of an extensive collection of tweets on the disease. 61,379 tweets that were posted on Twitter between May 7 and June 11, 2022, make up the dataset [26].These tweets show a variety of public sentiments on monkeypox, including neutral, negative, and favorable views. All tweets are deemed pertinent to debates about monkeypox in this study, offering a wide range of user viewpoints. To make sure the dataset is clean, balanced, and organized for analysis, pre-processing is an essential step before beginning any classification or prediction activities. The gathered raw tweets are naturally disorganized and include superfluous parts like stop words, duplicate records, and non-standardized content. Because Twitter is an unstructured medium with multilingual support, careful data pre-processing is necessary to get relevant results. A two-step procedure is used to eliminate duplicate records at the start of the pre-processing pipeline. Initially, the "is retweets" characteristic that Twitter gave was used to find duplicates. Then, based on their distinct tweet IDs and content, repetitive tweets were removed. After that, tweets were cleaned up to eliminate unnecessary content:

- Elements including URLs, email addresses, hash tags, mentions, and numerical data were removed using TextBlob analyzers.
- To ensure uniformity, all text was changed to lowercase, and stop words and punctuation were eliminated to concentrate on the important information.
- To maintain consistency in the linguistic research, only tweets written in English were kept.

The dataset is now well-structured and prepared for semantic analysis after being cleaned to remove noise and unnecessary information. The dependability of ensuing categorization and prediction tasks is guaranteed by this thorough pre-processing. Furthermore, the dataset utilized in this study is openly accessible, which encourages transparency and makes it possible for the research findings to be replicated.

2.2. *Create meaningful links between entities*

Creating meaningful links between entities is a crucial step in semantic analysis, particularly when dealing with unstructured or semi-structured data such as social media tweets. This process transforms disorganized datasets into structured knowledge, enabling deeper insights and efficient data interpretation. The primary objective of linking entities is to uncover relationships, organize fragmented information, and support analytical applications such as sentiment analysis and trend monitoring. For instance, tweets about monkeypox may mention various entities like "Human Monkeypox," "Zoonotic Disease," and "Monkeypox Virus." Establishing meaningful connections among these entities allows for better understanding and representation of the data, making it more accessible for analysis. Graph theory plays a central role in this process by structuring data as a network of nodes and edges, where nodes represent entities and edges denote their relationships. Knowledge Graphs (KGs) [27], a practical application of graph theory, facilitate this transformation by encoding information as semantic triplets. These triplets, such as ("Human Monkeypox", "is a", "Zoonotic Disease") or ("Zoonotic Disease", "caused by", "Monkeypox Virus"), represent real-world knowledge in a machine-readable format. KGs are particularly useful for organizing complex datasets, enabling the discovery of hidden relationships and improving the scalability of data analysis. In the context of monkeypox-related tweets, creating meaningful links helps structure the data, enabling accurate sentiment analysis, misinformation detection, and trend identification. Techniques like entity extraction, relationship identification, and graph construction form the foundation of this process. By linking entities based on their semantic relationships, the data becomes more coherent, allowing AI models to perform more effective reasoning and prediction. This structured representation is not only vital for understanding public sentiment but also aids in healthcare analytics by improving data representation and enabling knowledge inference. Ultimately, the integration of graph theory into entity linking enhances the ability to extract meaningful insights from unstructured data, supporting data-driven decision-making in public health and other domains.

When v represents the set of B nodes, $|v| = B$; εrepresents the set of edges linking these nodes, and Z is the adjacency matrix, a graph may be summed up as follows: $J = (\nu,\ \varepsilon,\ z)$. The networks between any two nodes in v are designated by the adjacency matrix, where the entry of Z in the h-th row and g-th column indicates the significance of the link between the h-th and g-th nodes, and is represented as z_{hg}. The convolution action for spectral-based KG is defined in the Fourier domain by calculating the graph Laplacian Eigen decomposition. The graph that has been normalized Laplacian is definite as (D is the graph's degree matrix and A is its adjacency matrix), where Λ is a diagonal matrix containing its eigenvalues and the columns of U are its eigenvector matrix.

$$j_\theta{}^{*}p = u_{j_\theta}(\wedge)u^{S}p \tag{1}$$

A Chebyshev polynomial $S_a(p)$of instruction m appraised at $\widetilde{l}$is recycled, and the action is definite as

$$j_\theta{}^{*}p \approx \sum_{a=0}^{A-1}\theta_a S_a(\widetilde{l})p \tag{2}$$

where$\widetilde{l}$ is the diagonal scale matrix. The graph fusion layer in KG combines information from multiple vertices into a single vertex, which reduces the size of the graph and expands the acceptance field of graph filters [28]. To alleviate the problem of overestimating the local neighborhood structure of maps with a very wide node size distribution, the convolutional filter is reduced in size to $K = 1$ and approximated by λ ≈ 2,

$$j_\theta{}^{*}p \approx \theta'_0 p + \theta'_1 p(l - H_B)p = \theta'_0 p + \theta'_1 C^{-1/2}MC^{-1/2}p \tag{3}$$

Here,θ'_0, $\theta'_1$are two unimpeded variables. To restrain the number of limitations and avoid over fitting, KG further assume that$\theta = \theta'_0 - \theta'_1$, leading to the subsequent description of a graph convolution as follows.

$$j_\theta{}^{*}p \approx \theta\left(H_B + C^{-1/2}MC^{-1/2}\right)p \tag{4}$$

The definition of a signal$P \in r^{BPf}$ with C input networks and F filters for functional mapping is as surveys:

$$W = \widetilde{C}^{-1/2}\widetilde{M}\widetilde{C}^{-1/2}P\Theta \tag{5}$$

where$\Theta \in r^{DPf}$the matrix is generated by the filter bank parameters, and$W \in r^{BPf}$ is the signal matrix attained by difficulty. GraphSAGE is spatial-GCN that uses node implanting with maximum union combination. In order to save memory while sacrificing time performance, the authors propose a block training algorithm for GCNs. The GraphSAGE framework builds embeddings by selecting and combining features from the local neighborhood of a node.

$$\begin{aligned} i^{s}_{B_v} &= aggregate_s\left(\left\{i^{s-1}_{U},\ \forall U\ \in\ B_v\right\}\right), \\ i^{s}_{V} &= \sigma\left(Z^{s}\cdot\left[i^{s-1}_{V}\|i^{s}_{B_v}\right]\right) \end{aligned} \tag{6}$$

where$B_v$the area set of node is$i^{s}_{V}$is the concealed state of node V at time step S, and Z^s is the heaviness matrix at layers. Finally, K represents vector concatenation and σ is the logistic sigmoid function. The following is a formulation of the focus mechanism:

$$U_s = \text{tan}i(Zi_s + n) \tag{7}$$

$$\alpha_s = \frac{Exp\left(U^{S}_{s}U_z\right)}{\sum_{g=1}^{b}Exp\left(U^{S}_{s}U_z\right)} \tag{8}$$

$$T_s = \sum_{s}\alpha_s i_s \tag{9}$$

wherei_s is the production of every deposit; Z, U_z, and n are trainable masses and bias. The position of every component in i_s is unhurried by appraising the compilation between U_s and i_s, which is randomly prepared. αt is a SoftMax function. A graph courtesy network by stacking a single graph devotion deposit, a, which is a single-layer feed forward neural network, parameterized by a weight vector$\overrightarrow{m} \in r^{2fh}$. The layer figures the constants in the consideration devices of the node pair (h, g) by

$$\alpha_{h,g} = \frac{Exp\left(leakyrelu\left(\overrightarrow{m}^{S}\left[Z\overrightarrow{i}_{h}\|\ Z\overrightarrow{i}_{g}\right]\right)\right)}{\sum_{K\in B_hB}Exp\left(leakyrelu\left(\overrightarrow{m}^{S}\left[Z\overrightarrow{i}_{h}\|\ Z\overrightarrow{i}_{g}\right]\right)\right)} \tag{10}$$

where || represents the chain operation. The courtesy layer takes as input a set of node features$i = \left\{\overrightarrow{i_1},\ \overrightarrow{i_2},\ \ldots,\ \overrightarrow{i_B}\right\}$, $\overrightarrow{i_1} \in r^{f}$, where B is the number of protuberances of the input graph and f the number of structures for every node, and foodstuffs set of node features $i' = \left\{\overrightarrow{i_1}',\ \overrightarrow{i_2}',\ldots,\ \overrightarrow{i_B}'\right\}$, $\overrightarrow{i_1}' \in r^{f}$as its output. The first stage in creating higher-level features is to apply a common linear transformation to each node, which is parameterized by a weight matrix $Z \in r^{f'*f}$. Each node may then be subjected to a masked attention mechanism, which yields the following scores:

$$E_{hg} = m\left(X\overrightarrow{i}_{h}, X\overrightarrow{i}_{g}\right) \tag{11}$$

Which specifies the position of node g's features to node i. A nonlinearity, σ, can be applied to each node to obtain its final output feature.

$$i'_h = \sigma\left(\sum_{g\in B_h}\alpha_{hg}Zi_g\right) \tag{12}$$

In order to stabilize the learning process, the layer additionally employs numerous attentions. The following representation is produced by combining the individual characteristics that are computed in parallel by k distinct nodes:

$$i'_h = \|_{k=1}^{k} \sigma\left(\sum_{g \in B_h} \alpha_{hg}^{K} Z^{K} \overrightarrow{i}_g\right) \tag{13}$$

By retaining be an average of and delay applying the final nonlinearity.

$$i'_h = \sigma\left(\frac{1}{K} \sum_{K=1}^{K} \sum_{j \in B_h} \alpha_{hg}^{K} Z^{K} \overrightarrow{i}_g\right) \tag{14}$$

where$\alpha_{hg}^{K}$is the standardized consideration coefficient compute by the k-th attention mechanism.

2.3. Sentiment analysis to detect opinions on monkeypox infection

Once meaningful links between entities are created using Knowledge Graphs, the process of sentiment classification begins. Each tweet is analyzed to determine whether the sentiment expressed is positive, negative, or neutral. This study employs a Reinforcement Markov Decision Process (RMDP) [29] to enhance sentiment analysis by treating it as a sequential decision-making problem. The process begins with preprocessing and feature extraction using techniques like GloVe embeddings and one-hot encoding, which capture the syntactic and semantic nuances of the text. These features are then passed through the RMDP framework, where sentiment analysis is treated as a series of actions within a structured decision-making environment. The Markov decision process (MDP) provides the foundational framework for reinforcement learning by modeling the problem as a set of states, actions, rewards, and transitions. In this context, the states represent tweet features, the actions correspond to sentiment classifications (positive, negative, or neutral), and the rewards signify the accuracy of classifications based on ground truth. RMDP-based sentiment analysis not only improves classification accuracy but also provides a robust and adaptive methodology for analyzing public sentiment around monkeypox infection in social media, offering valuable insights to guide health communication strategies. Then, depending on the state changeover probabilities$X_{t_s,t_{s+1}}(m_s)$, the situation evolves to a new state. For this development, the agent is immediately rewarded withR_s. The agent's objective is to maximize the expected cumulative advertising reward over time by obtaining a policy $\pi(m_s|t_s)$ that associates each state t_s with an accomplishmentm_s.

$$Y^{\pi}(t_s, m_s) = e^{\pi}[R_s | t_s, m_s] \tag{15}$$

For any state action pairings (t, m), a policy π* is considered the best if and only if its projected payoff is greater than or equal to π.

$$Y^{\pi^*}(t, m) \geq Y^{\pi}(t, m) \tag{16}$$

The ultimate objective of an MDP is the Y^{π^*} function, which describes the highest anticipated reward achievable by carrying out a certain action in a specific condition and then following the best course of action.

$$Y^*(t_s, m_s) = e_{t_{s+1} \sim X_{t_s,t_{s+1}}(m_s)}\left[r(t_s, m_s) + \gamma \underset{m' \in M}{Max} Y^*(t_{s+1}, m')\right] \tag{17}$$

where $r(t_s, m_s)$is the instantaneous reward gotten after executing action m_s in state$t_s$at time s, $t_{s+1}$is the next state, $m'$is any achievement that can be busy at t_{s+1}, and γ is a markdown factor that regulates the weight agreed to coming prizes. Conferring to the value of active software design can be written as follows:

$$Y^*(t_s, m_s) = Max\{r(t_s, m_s) + \gamma \sum_{m' \in M} X_{t_s, t_{s+1}}(m_s) Y^*(t_{s+1}, m')\} \tag{18}$$

Since the transition prospects of states ($X_{t_s,t_{s+1}}(m_s)$) and the optimal payoffs of posterior sub-processes ($Max_{m' \in M} Y^*(t_{s+1}, m')$)are often unknown. When given some experience following a policy π, the sentiment analysis updates the estimate $Y(t_s, m_s)$ for the non-terminal states t_s occurring in that experience. The simplest bring up-to-date rule for the opinion detection is as follows:

$$Y(t_s, m_s) \leftarrow Y(t_s, m_s) + \alpha[R_{s+1} + \gamma Y(t_{s+1}, m_{s+1}) - Y(t_s, m_s)] \tag{19}$$

where α is the erudition rate. Signifies an error measures the difference among the value of $Y(t_s, m_s)$ and the better estimation$R_{s+1} + \gamma Y(t_{s+1}, m_{s+1})$. This number is frequently referred to as the false alarm, and arises in various forms throughout reinforcement learning.

$$\delta(s) \doteq R_{s+1} + \gamma Y(t_{s+1}, m_{s+1}) - Y(t_s, m_s) \tag{20}$$

The RMDP consists of two networks [30]: the current network with parameters ω and the target network with parametersω'. On the other hand, the target network is used to estimate the target Y value which guides the training process. The present network's parameters are regularly transferred to the target network at intervals off epochs.

$$l(\omega) = e_{(t,m,r,t') \sim u(C)}[(Y_{tar}(t', m', \omega') - Y_{tar}(t, m, \omega))]^2 \tag{21}$$

$$\nabla_{\omega} l(\omega) = e_{(t,m,r,t') \sim u(C)}[l(\omega)] \tag{22}$$

where(t, m, r, t$'$) ∼ u(C)specifies that data (t, m, r, t$'$)is sampled from involvement replay pool C, and define the objective functions as follows.

$$l'(\omega) = (Y_{tar}(t', m', \omega') - Y_{tar}(t, m, \omega))\nabla_{\omega} Y(t, m, \omega) \tag{23}$$

Instead of randomly selecting the optimal action based on the Y function, the agent randomly selects actions with a fixed probability at each step. If the random number ξ generated by the algorithm is less than the search probability calculated in the power greedy algorithm, then the current optimal solution is based on the objective function (Y) is $\frac{1}{|M|}$selected by probability where |m| indicates the size of the working space [30]. Otherwise, the current optimal action based on the Y function is selected with probability.

$$x(m^*|t) = \begin{cases} \frac{1}{|M|} & if\ \xi < \in \\ 1 & if\ \xi < \in \end{cases} \tag{24}$$

wherem^*, Y, |M| the current ideal function of the purpose is the size of the functional space and 0 ≤ ξ ≤ 1 is a even random number. The Softmax system uses the Boltzmann distribution to estimate the Y values of actions and determine the probability of selecting an action.

$$x(m_h|t) = \frac{E^{\frac{Y(t, m_h)}{\tau}}}{\sum_{g=1}^{|M|} E^{\frac{Y(t, m_g)}{\tau}}} \tag{25}$$

where $h = 1, 2, 3, \ldots,$ |M|,$x(m_h|t)$ is the probability of choosing an action in state t$Y(t, m_h)$, the expected value Y of the action $m_h$in state t, τ is the temperature parameter, and |M| Size of working space.

$$F(p, q) = \frac{1 - E^{\frac{-|Y_{tar}(t, m, \omega') - Y(t, m, \omega)|}{\sigma}}}{1 + E^{\frac{-|Y_{tar}(t, m, \omega') - Y(t, m, \omega)|}{\sigma}}} \tag{26}$$

where$Y_{tar}(t, m, \omega')$is the Y charge for the mark net trim state act pair (t, m) and y(t, m, ω) is the Y worth for the network close-fitting state action pair (t, m). ω'And ω are the limitations of the goal and existing systems, individually, and σ is a optimistic persistent called the inverse compassion factor. As stated in the earlier section, the false alarm is usually articulated as δ(s).

$$j(\delta(s), \sigma) = \frac{1 - E^{\frac{-|\delta(s)|}{\sigma}}}{1 + E^{\frac{-|\delta(s)|}{\sigma}}} \tag{27}$$

The value of σ the function$j(\delta(s), \sigma)$ is not monotonically decreasing with respect to δ(s) and its value lies in the interval [0, 1). The probet_{s-1} returns the probability ∈ (t_{s-1})of state s - 1. where β ∈ (0, 1), typically 1/|M|, which defines the effect of TD error on detection probability, where |M| Indicates the size of the working space.

$$\in (t_s) = (1-\beta)\cdot j(\delta(s), \sigma) + \beta\cdot \in (t_{s-1}) \tag{28}$$

It is designed to optimize an agent's search strategy in an RL system. The search probability is constantly modified depending on the agent's present environmental information by including TD error into the softmax algorithm. The difference between the Y values derived from the target network and the present network is measured by the TD error.

3. Results and discussion

This section presents the results and comparative analysis of sentiment analysis for opinion detection in monkeypox. Python programming is used to do experiments on an X86–64 Ubuntu 18.04.4 LTS computer. With 16 GB of RAM, the CPU is an Intel(R) Core(TM) i7–8550 U running at 1.80 GHz. Using the suggested model, we examine sentiment emotions using this configuration. The Monkeypox Twitter dataset used in this study was sourced from the open-source platform GitHub. It includes a comprehensive collection of tweets related to the disease, comprising 61,379 tweets posted on Twitter between May 7 and June 11, 2022 [31]. These tweets capture a range of public sentiments regarding monkeypox, including neutral, negative, and positive opinions. All the tweets are considered relevant to the discussions surrounding monkeypox, providing diverse perspectives from users. The results of proposed Graph+RMDP model is compared with the existing models such as Navie bayes (NB), support vector machine (SVM), logistic regression (LR), random forest (RF), decision tree (DT), convolutional neural network (CNN), long short-term memory (LSTM) and CNN+LSTM [32]. The performance can validated through different metrics such as accuracy, precision, recall, F-measure and area under curve (AUC).

3.1. Impact of graph theory in sentiment analysis for opinion detection in monkeypox

The integration of graph theory in sentiment analysis offers a powerful method for understanding complex relationships within large datasets, such as social media discussions surrounding events like the monkeypox outbreak. This approach enhances the ability to detect public opinion, categorize sentiments accurately, and uncover deeper insights into societal reactions. Graph theory provides a structured way to organize unstructured social media data. Tweets about monkeypox are often disjointed and varied in content, making it difficult to extract meaningful relationships between entities (e.g., symptoms, prevention, public health policies). By using graphs, each tweet or piece of information can be treated as a node, and relationships between these nodes—such as co-occurrence of terms or sentiment-related associations—can be established as edges. This allows the creation of a network (Fig. 2) that illustrates how different concepts (e.g., fear, misinformation, and vaccination) are interconnected, facilitating deeper semantic analysis.

Traditional sentiment analysis techniques typically classify text into predefined sentiment categories (positive, negative, or neutral). However, this often oversimplifies complex opinions that may contain mixed emotions or contradictory viewpoints. Graph theory helps address this by identifying patterns of sentiment propagation across the network. For example, the sentiment of an individual tweet about monkeypox may influence subsequent tweets, either amplifying or modifying the general public's perception. In the context of monkeypox, misinformation and rumors can spread rapidly, exacerbating panic and confusion. Graph theory enhances misinformation detection by analyzing the flow of information through the network and identifying unusual patterns that may indicate false narratives. For instance, a rapid increase in the spread

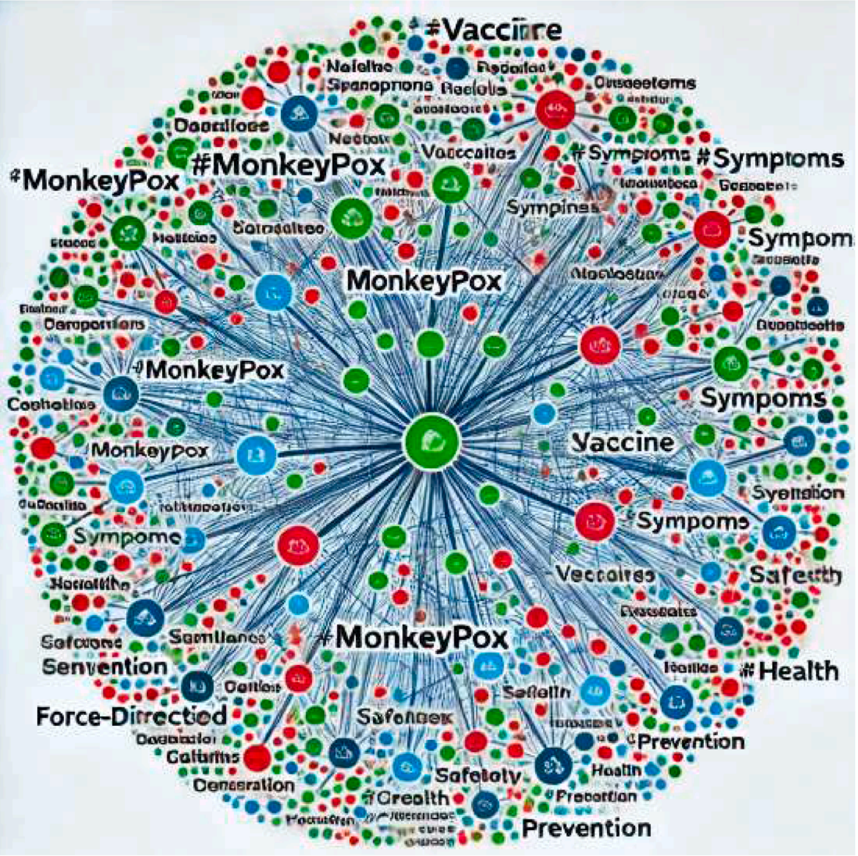


Fig. 2. Knowledge graph for limited tweets from Monkeypox tweet dataset, the plot highlights positive (green), negative (red), and neutral (blue) sentiments. The graph shows the relationships between keywords, hashtags, and user interactions, such as "monkeypox", "vaccine", "symptoms", "health", and "prevention".

of a specific hashtags associated with misinformation could be detected by examining the graph's connectivity and temporal patterns. Sentiment propagation models using graph theory take into account the influence of neighboring nodes in a network. When a tweet expressing a strong sentiment spreads through a network, the sentiment of connected nodes may also shift. Fig. 3visually represents the relationships between key entities and sentiment in the monkeypox tweet dataset. By applying graph-based algorithms, sentiment analysis can account for this dynamic process, leading to a more accurate and nuanced understanding of how sentiment evolves over time and across different demographic groups. This allows for timely interventions and better-targeted public health campaigns regarding monkeypox. By combining sentiment analysis with graph theory, researchers can contextualize public opinion in ways that are more meaningful for decision-makers. For example, graph-based sentiment analysis help identify which areas or regions are most concerned about monkeypox, the type of concerns and how these concerns evolve over time. It enables health authorities to tailor their communication strategies, target high-risk groups, and address misconceptions in a more informed and efficient manner.

3.2. Results analysis of sentiment analysis for monkeypox tweets

This section presents a comparative analysis of the proposed Graph+RMDP model and several existing models, including NB, SVM, LR, RF, DT, CNN, LSTM and CNN+LSTM [32], for sentiment analysis to detect public opinion on monkeypox tweets. The performance of these models is evaluated using various metrics, including accuracy, precision, recall, F-measure, and AUC. Fig. 4 shows the results of the proposed Graph+RMDP model indicate effective learning and generalization as seen in both the loss and accuracy curves across the epochs. The train loss starts at 0.56 in epoch 0 and consistently decreases, reaching a minimal value of 0.0000002 by epoch 59. This steady reduction in train loss suggests that the model is effectively learning from the training data, improving its predictions over time. The test loss follows a similar downward trend, starting at 0.59 and decreasing to 0.17 by epoch 59, albeit with some fluctuations. The test loss is slightly higher than the train loss throughout, which is typical in machine learning models, indicating a minor degree of overfitting. The train accuracy shows rapid improvement, starting at 0.6 in epoch 0 and reaching 1.0 by epoch 20. After this point, it stabilizes at a perfect score, suggesting that the model is fitting the training data very well. In contrast, test accuracy starts at 0.55 and gradually increases, peaking at 0.99 by epoch 23. Although it fluctuates slightly between 0.94 and 0.99 after this point, the general upward trend in test accuracy indicates that the model is not just memorizing the training data but is also able to generalize well to new, unseen data. The fluctuations in test accuracy, especially between epochs 27 to 37, where it dips to 0.95, could be due to minor overfitting or noise in the validation set, but the overall high values demonstrate that the model performs reliably on test data. The Graph+RMDP model shows excellent performance, with a steady decline in both train and test loss and high, stable test accuracy. Although the model achieves perfect training accuracy, the test accuracy fluctuates slightly, indicating a small degree of overfitting. These results suggest that the model is effectively learning from the data, with good generalization ability, though further refinements or regularization techniques could be explored to reduce the minor fluctuations in test accuracy.

Table 2 summarizes the performance analysis of the proposed and existing sentiment analysis models for the Monkeypox tweet dataset. In the accuracy comparison for the Monkeypox tweet dataset (Fig. 5), all

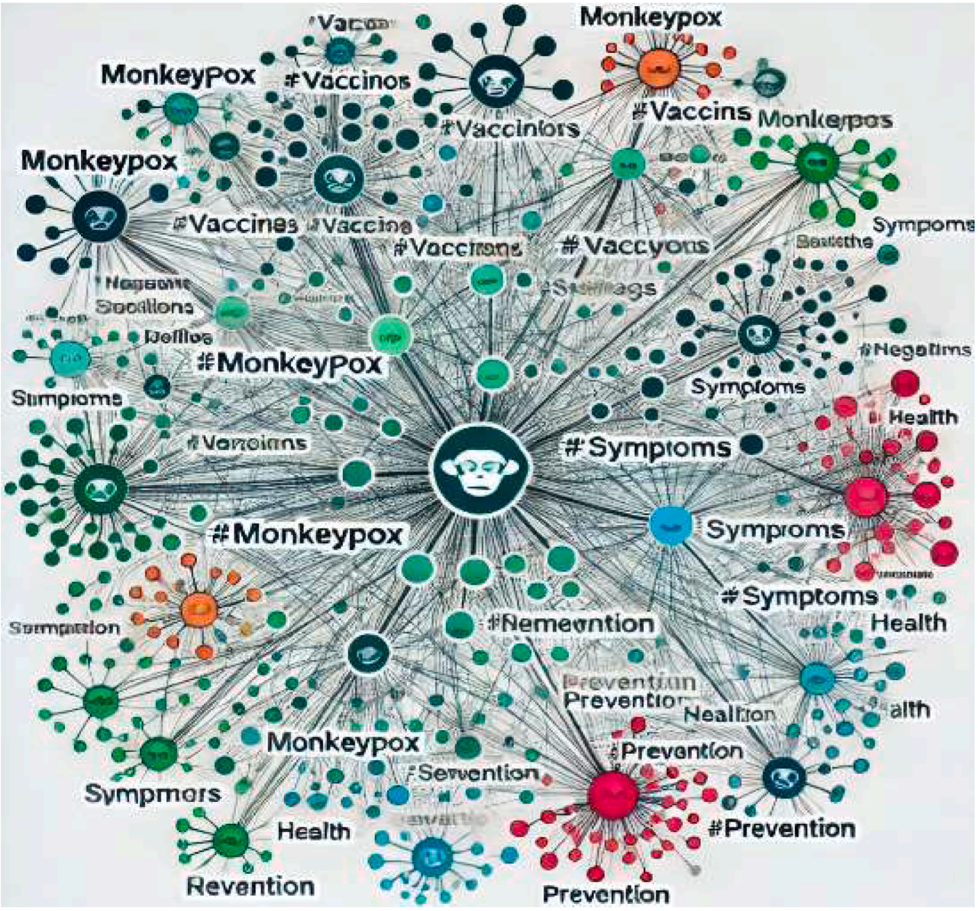


Fig. 3. Knowledge graph for entire Monkeypox tweet dataset.

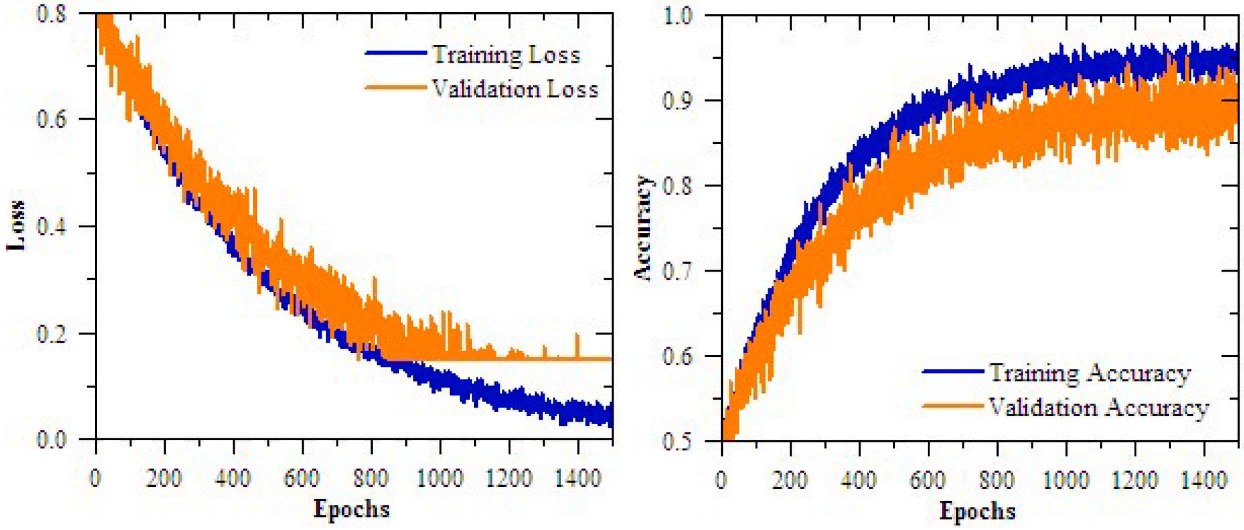


Fig. 4. Loss and accuracy of proposed Graph+RMDP model with varying epochs.

Table 2
Performance analysis of proposed and existing sentiment analysis models for Monkeypox tweet dataset.

Models	Accuracy (%)					Precision (%)				
	300	600	900	1200	1500	300	600	900	1200	1500
NB	75.236	75.858	78.958	81.256	84.578	72.526	73.112	76.436	77.104	80.302
SVM	78.528	79.158	81.254	82.366	85.546	75.307	77.348	79.451	80.903	82.739
LR	80.147	82.355	86.645	89.578	90.125	78.109	78.723	81.286	83.301	85.220
RF	84.153	85.158	86.985	90.124	90.857	80.414	81.752	84.240	85.234	87.518
DT	85.633	86.247	87.985	91.985	92.357	81.201	82.254	84.853	86.053	88.353
CNN	89.547	90.058	91.547	92.285	93.398	83.673	84.381	87.110	89.022	90.441
LSTM	90.258	91.086	92.357	93.325	94.475	85.420	86.539	88.211	89.441	91.012
CNN+LSTM	95.628	95.857	95.957	96.012	96.235	88.119	88.986	90.407	91.228	92.731
Graph+RMDP	98.564	98.618	98.855	98.958	99.245	93.212	94.451	95.640	96.125	97.076
	Recall (%)					F-measure (%)				
NB	70.345	71.850	74.491	75.693	78.204	71.419	72.476	75.451	76.392	79.239
SVM	73.270	74.634	77.122	78.479	81.017	74.275	75.967	78.269	79.673	81.869
LR	75.301	76.429	79.110	80.563	82.354	76.679	77.559	80.183	81.909	83.762
RF	77.028	78.024	80.136	81.543	82.953	78.685	79.845	82.137	83.348	85.174
DT	78.712	79.324	82.139	83.127	85.043	79.937	80.762	83.474	84.565	86.666
CNN	80.210	81.352	84.016	85.178	87.322	81.905	82.839	85.535	87.058	88.854
LSTM	82.145	83.509	85.431	86.450	88.360	83.750	84.997	86.799	87.920	89.666
CNN+LSTM	85.235	85.951	88.024	89.301	90.702	86.653	87.442	89.200	90.254	91.705
Graph+RMDP	90.124	91.016	92.410	93.047	94.310	91.642	92.702	93.997	94.561	95.673

models show improved performance as the dataset size increases from 300 to 1500 epochs. The NB model improves from 75.236 % to 84.578 %, SVM from 78.528 % to 85.546 %, LR from 80.147 % to 90.125 %, RF from 84.153 % to 90.857 %, and DT from 85.633 % to 92.357 %. Deep learning models like CNN and LSTM also perform better, with CNN improving from 89.547 % to 93.398 %, and LSTM from 90.258 % to 94.475 %. The CNN+LSTM hybrid model performs even better, increasing from 95.628 % to 96.235 %. The Graph+RMDP model consistently achieves the highest accuracy, improving from 98.564 % to 99.245 %. This superior performance is attributed to its ability to capture complex relationships between tweet features using graph-based learning, while the RMDP component enhances decision-making by dynamically optimizing classification outcomes based on long-term reward signals. Unlike traditional models that rely on static feature vectors, Graph+RMDP offer a more adaptive and context-aware sentiment analysis approach, making it highly effective for dynamic and noisy social media data.

Fig. 6 presents the precision comparison of the proposed and existing models on the Monkeypox tweet dataset. The Graph+RMDP model achieves the highest precision across all dataset sizes, starting at 93.212 % and reaching 97.076 %, reflecting a 3.86 % improvement. The CNN+LSTM model also shows strong performance, improving from 88.119 % to 92.731 % (4.61 % increase), followed by LSTM and CNN with gains of 5.59 % and 6.77 %, respectively. Among traditional models, DT improves from 81.201 % to 88.353 % and RF from 80.414 % to 87.518 %, showing steady gains. NB starts at 72.526 % and improves to 80.302 % (7.78 % increase), while SVM and LR record improvements of 7.43 % and 7.11 %, respectively. The results highlight the effectiveness of advanced models—particularly Graph+RMDP and CNN+LSTM—in delivering superior precision in sentiment analysis of Monkeypox-related tweets, outperforming traditional approaches by a considerable margin. Fig. 7 illustrates the recall comparison of the proposed and existing models on the Monkeypox tweet dataset. The Graph+RMDP model consistently achieves the highest recall, increasing from 90.124 % at 300 epochs to 94.31 % at 1500 epochs, marking 4.19 % improvement. The CNN+LSTM model follows, improving from 85.235 % to 90.702 % (5.47 % increases). Similarly, LSTM and CNN show notable gains of 6.22 % and 7.11 %, respectively, across the dataset sizes. Among traditional models, DT and RF show steady improvements, with DT rising by 6.33 % and RF by 5.92 %. LR and SVM

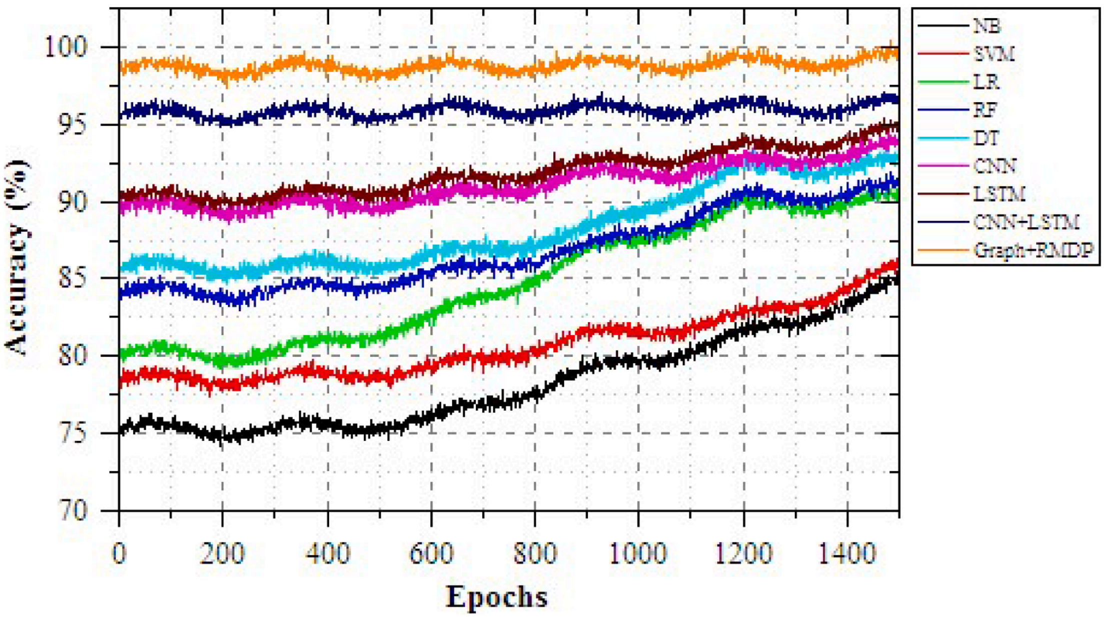


Fig. 5. Accuracy comparison for proposed and existing models for Monkeypox tweet dataset.

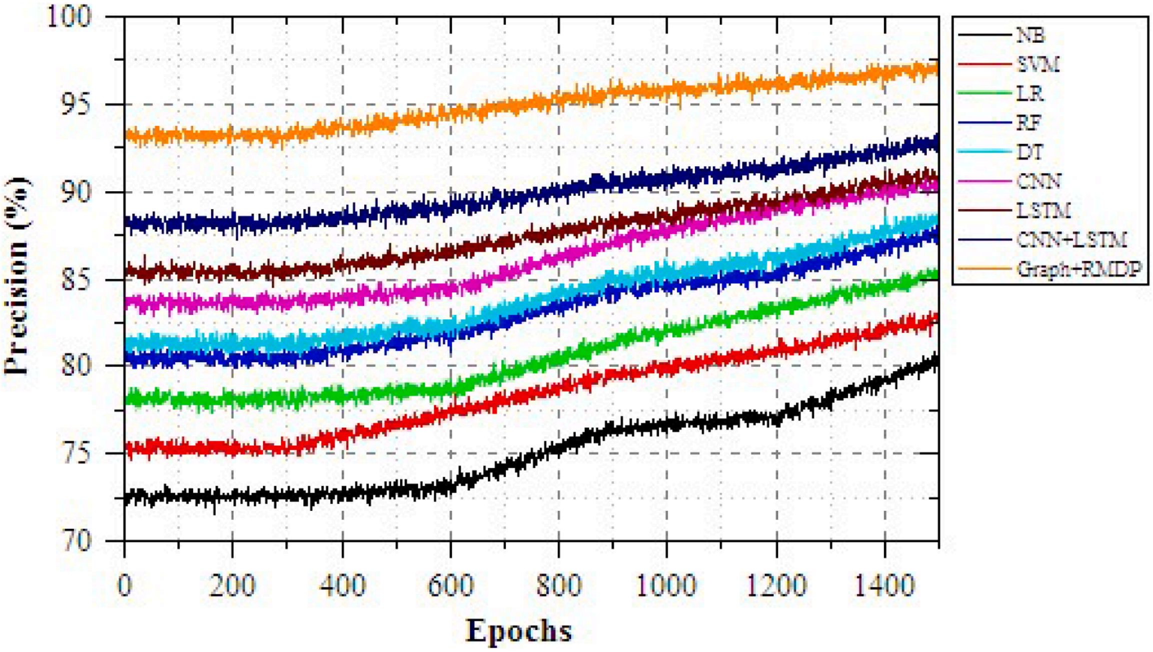


Fig. 6. Precision comparison for proposed and existing models for Monkeypox tweet dataset.

demonstrate moderate growth, with recall increasing by 7.05 % and 7.75 %, respectively. Although NB shows the smallest improvement (7.86 %), it performs relatively well at lower dataset sizes. The results highlight the superior recall performance of advanced models like Graph+RMDP and CNN+LSTM, while traditional models also exhibit consistent gains with increased dataset size.

Fig. 8 presents the F-measure comparison of the models on the Monkeypox tweet dataset. The Graph+RMDP model achieves the highest F-measure, improving from 91.642 % at 300 to 95.673 % at 1500 epochs, reflecting a 4.03 % increase. The CNN+LSTM model follows closely, rising from 86.653 % to 91.705 %, marking 5.05 % improvement. Similarly, LSTM and CNN show notable gains of 5.92 % and 6.95 %, respectively. Among the traditional models, DT and RF exhibit steady performance, with DT improving by 6.73 % and RF by 6.49 %. LR and SVM show moderate increases of 7.08 % and 7.59 %, respectively. The NB model shows the smallest gain, increasing by 7.82 %, from 71.419 % to 79.239 %. The results indicate that hybrid models like Graph+RMDP and CNN+LSTM outperform traditional approaches, offering improvements in F-measure, particularly as dataset size increases.

Fig. 9 shows the performance of various sentiment analysis models for opinion detection on the Monkeypox tweet dataset was assessed across five key metrics: accuracy, precision, recall, F-measure, and AUC. The Graph+RMDP model emerged as the top performer, achieving the highest accuracy of 98.848 %, which was 2.91 % improvement over CNN+LSTM. Compared to traditional models, Graph+RMDP showed a significant increase in accuracy, with models like NB lagging far behind by 19.671 %. The precision of Graph+RMDP was also the highest at

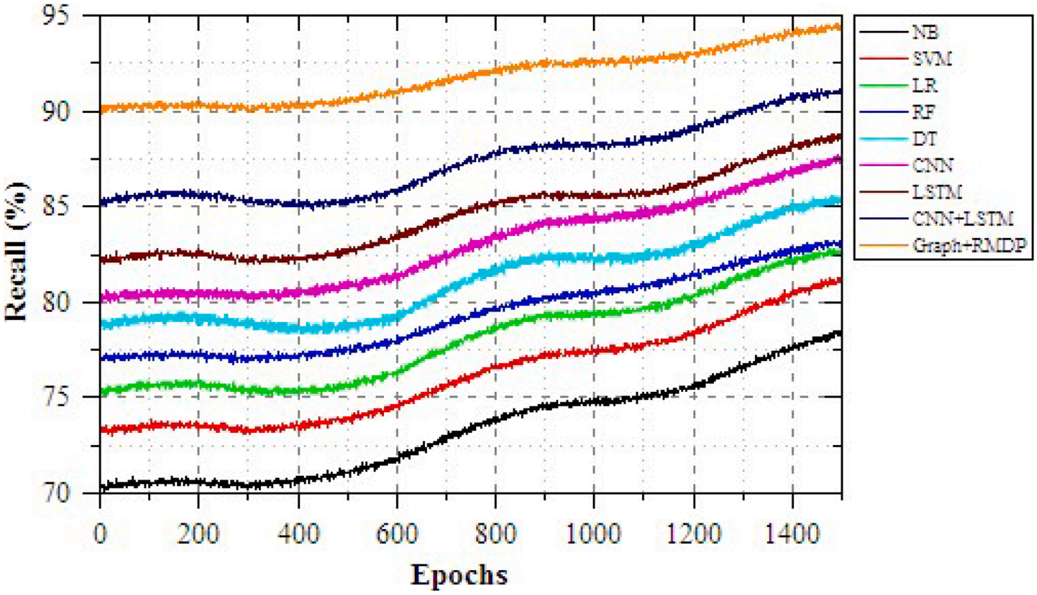


Fig. 7. Recall comparison for proposed and existing models for Monkeypox tweet dataset.

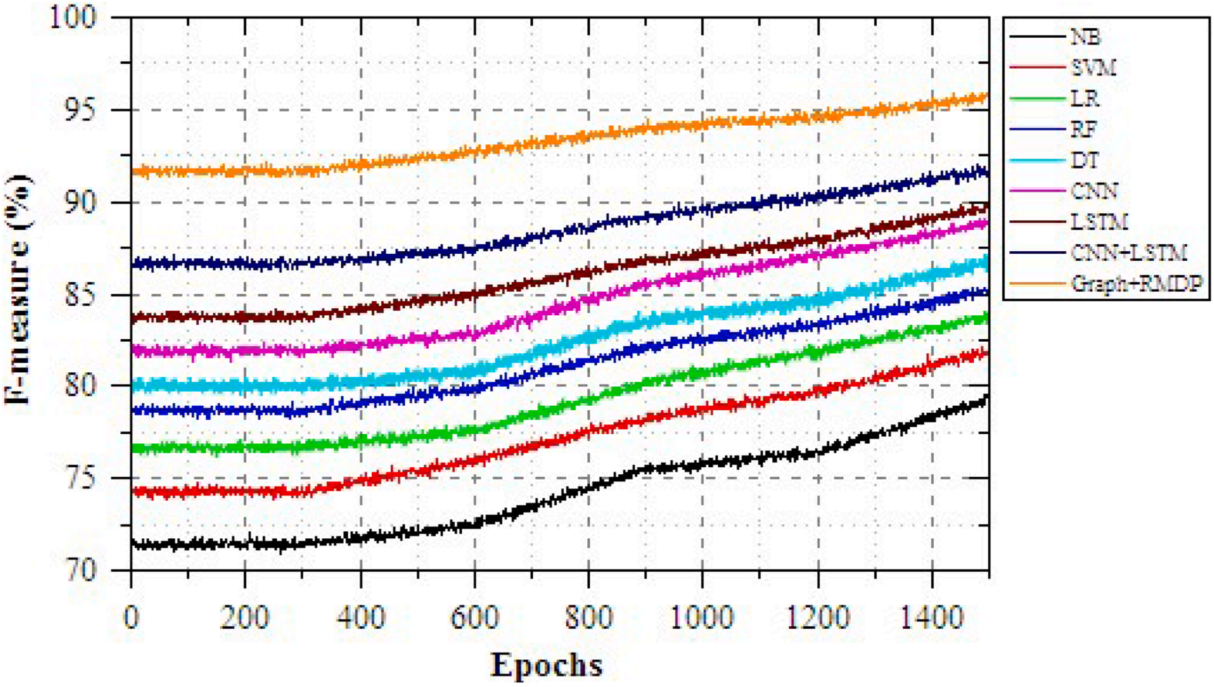


Fig. 8. F-measure comparison for proposed and existing models for Monkeypox tweet dataset.

95.301 %, exceeding CNN+LSTM by 5.007 %, with NB again showing the lowest precision at 75.896 %, a decrease of 19.405 % compared to the top model. In terms of recall, Graph+RMDP led with a score of 92.181 %, outperforming CNN+LSTM by 4.057 %, and traditional models like RF and DT showed lower performance. NB had the lowest recall at 74.117 %, trailing the best model by 18.064 %. For F-measure, Graph+RMDP also achieved the highest score of 93.715 %, surpassing CNN+LSTM by 4.664 %. This marked an impressive improvement over models like RF and DT with NB again showing the weakest performance with 74.995 %, falling behind by 18.72 %. The AUC scores mirrored these findings, with Graph+RMDP achieving 95.8 %, a 3.4 % improvement over CNN+LSTM. Traditional models like RF and DT exhibited lower AUC values, while NB again lagged behind with the lowest score of 78.45 %, a 17.35 % decrease compared to Graph+RMDP. Graph+RMDP consistently outperformed all other models across all metrics, showing significant improvements in accuracy, precision, recall, F-measure, and AUC. While CNN+LSTM also performed well, traditional ML models like NB, SVM, LR, RF, and DT demonstrated lower performance across the board.

4. Conclusion

The proposed AI-powered sentiment analysis, combined with graph theory, effectively addresses the challenges of analyzing unstructured or semi-structured social media data surrounding Monkeypox. By using graph theory to establish meaningful connections between keywords,

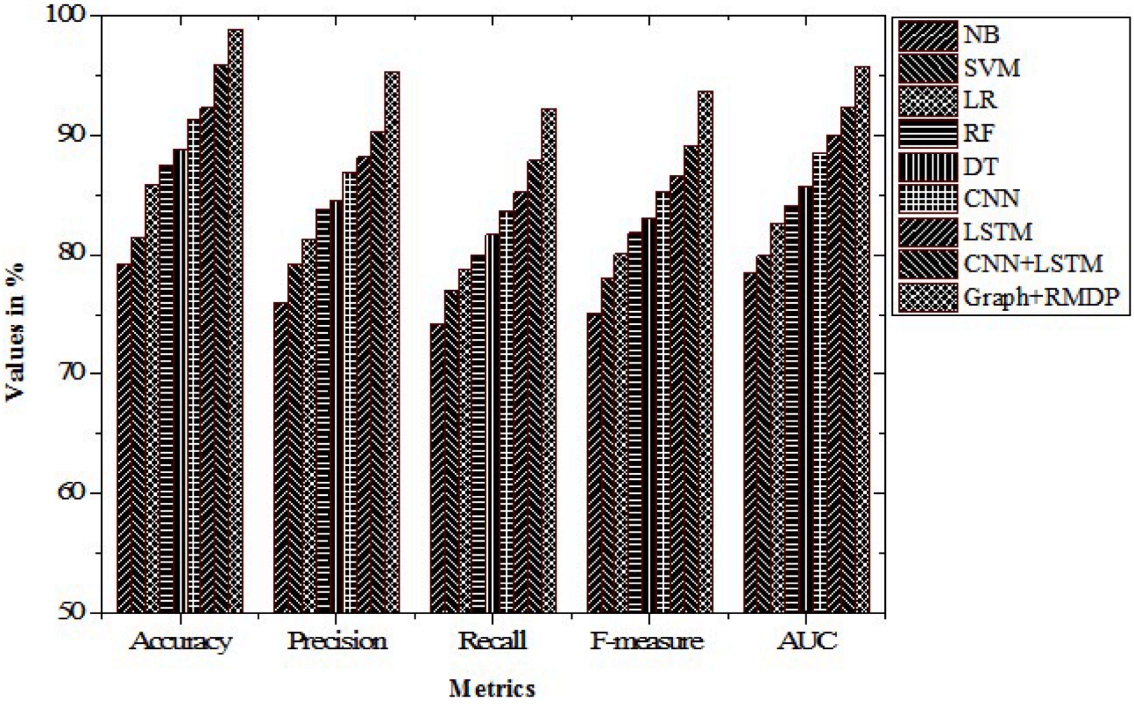


Fig. 9. Performance comparison of different models for sentiment analysis for opinion detection monkeypox.

hashtags, and user interactions, and using a reinforcement Markov decision process (RMDP) to analyze opinions and detect sentiment. The methodology was validated using a Monkeypox tweet dataset comprising 61,379 tweets collected from Twitter between May 7 and June 11, 2022.The results demonstrate that the Graph+RMDP model outperforms existing sentiment analysis models for the Monkeypox tweet dataset. It achieved the highest accuracy of 98.848 %, precision of 95.301 %, recall of 92.181 %, F-measure of 93.715 %, and AUC of 95.8 %, reflecting substantial improvements over the next best model, CNN+LSTM, with increases of 2.91 % in accuracy, 5 % in precision, 4.057 % in recall, 4.664 % in F-measure, and 3.4 % in AUC. When compared to traditional models such as Naïve Bayes, the Graph+RMDP model demonstrated a performance boost of up to 19.671 % in accuracy. The Graph+RMDP model is poised to provide valuable insights into public sentiment and trends related to public health crises like Monkeypox, thereby enabling more informed and data-driven decisions for policymakers and public health organizations.

Ethical approval

Not Applicable

Author contributions

All authors contributed equally to this work and discussed the results and implications and commented on the manuscript at all stages.

Funding

No funding were received from any research agency and organizations.

Availability of data and materials

Data available on reasonable request

CRediT authorship contribution statement

M. VENKATACHALAM: Writing – original draft, Visualization. **R. VIKRAMA PRASAD:** Writing – original draft, Validation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] N. Comeau, A. Abdelnour, K. Ashack, 42223 Assessing public interest in monkeypox via social media platforms: google Trends, YouTube, and TikTok, J. Am. Acad. Dermatol. 89 (3) (2023) AB123.

[2] R.A. Farahat, M.A. Yassin, J.A. Al-Tawfiq, C.A. Bejan, B. Abdelazeem, Public perspectives of monkeypox in Twitter: a social media analysis using machine learning, New Microbes New Infect 49 (2022).

[3] M. Baroudi, I. Smouni, H. Gourram, A. Labzai, M. Belam, Optimizing control strategies for monkeypox through mathematical modeling, Partial Diff. Eq. Appl. Math. (2024) 100996.

[4] K. Soni, A.K. Sinha, Modeling and stability analysis of the transmission dynamics of Monkeypox with control intervention, Partial Diff. Eq. Appl. Math. 10 (2024) 100730.

[5] J. Chire-Saire, A. Pineda-Briseño, J. Oblitas-Cruz, Sentiment analysis of monkeypox tweets in Latin America, in: International Conference on Applied Machine Learning and Data Analytics, Springer Nature Switzerland, Cham, 2023, pp. 230–245.

[6] S. Li, J. Chen, Virtual human on social media: text mining and sentiment analysis, Technol. Soc. 78 (2024) 102666.

[7] J. Du, J. Xu, H. Song, X. Liu, C. Tao, Optimization on machine learning based approaches for sentiment analysis on HPV vaccines related tweets, J. Biomed. Semantics 8 (2017) 1–7.

[8] Z.A. Khan, Y. Xia, A. Khan, M. Sadiq, M. Alam, F.A. Awwad, E.A. Ismail, Developing lexicons for enhanced sentiment analysis in software engineering: an innovative multilingual approach for social Media reviews, Comput. Mater. Contin. 79 (2) (2024).

[9] J. Yang, Y. Xiong, Social media sentiment contagion and stock price jumps and crashes, Pacific-Basin Finance J. 88 (2024) 102520.

[10] M. Ashayeri, N. Abbasabadi, Unraveling energy justice in NYC urban buildings through social media sentiment analysis and transformer deep learning, Energy Build 306 (2024) 113914.

[11] W. An, F. Tian, P. Chen, Q. Zheng, Aspect-based sentiment analysis with heterogeneous graph neural network, IEEe Trans. Comput. Soc. Syst. 10 (1) (2022) 403–412.

[12] L. Yuan, J. Wang, L.C. Yu, X. Zhang, Syntactic graph attention network for aspect-level sentiment analysis, IEEe Trans. Artif. Intell. 5 (1) (2022) 140–153.

[13] K. Liang, L. Meng, M. Liu, Y. Liu, W. Tu, S. Wang, S. Zhou, X. Liu, F. Sun, K. He, A survey of knowledge graph reasoning on graph types: static, dynamic, and multi-modal, IEEE Trans. Pattern Anal. Mach. Intell. (2024).

[14] X. Luo, S. Zhang, J. Wu, H. Chen, H. Peng, C. Zhou, Z. Li, S. Xue, J. Yang, ReiPool: reinforced pooling graph neural networks for graph-level representation learning, IEEE Trans. Knowl. Data Eng. (2024).

[15] M. Kalaimathi, B.J. Balamurugan, Topological indices of molecular graphs of monkeypox drugs for QSPR analysis to predict physicochemical and ADMET properties, Int. J. Quantum Chem. 123 (22) (2023) e27210.

[16] B. Yu, S. Zhang, A novel weight-oriented graph convolutional network for aspect-based sentiment analysis, J. Supercomput. 79 (1) (2023) 947–972.

[17] A.M. Schoene, L. Bojanić, M.Q. Nghiem, I.M. Hunt, S. Ananiadou, Classifying suicide-related content and emotions on Twitter using Graph Convolutional Neural Networks, IEEE Trans. Affect. Comput. 14 (3) (2022) 1791–1802.

[18] V. Shumovskaia, M. Kayaalp, M. Cemri, A.H. Sayed, Discovering influencers in opinion formation over social graphs, IEEe Open. J. Signal. Process. 4 (2023) 188–207.

[19] B.P. Nandi, A. Jain, D.K. Tayal, Aspect based sentiment analysis using long-short term memory and weighted N-gram graph-cut, Cognit. Comput. 15 (3) (2023) 822–837.

[20] Y. Zeng, Z. Li, Z. Chen, H. Ma, Aspect-level sentiment analysis based on semantic heterogeneous graph convolutional network, Front. Comput. Sci. 17 (6) (2023) 176340.

[21] H.T. Phan, N.T. Nguyen, A fuzzy graph convolutional network model for sentence-level sentiment analysis, IEEE Trans. Fuzzy Syst. (2024).

[22] W. Fu, S. Akbar, Expert profile identification from community detection on author-publication-keyword graph with keyword extraction, IEEe Access. (2024).

[23] J. Du, J. Jin, J. Zhuang, C. Zhang, Hierarchical graph contrastive learning of local and global presentation for multimodal sentiment analysis, Sci. Rep. 14 (1) (2024) 5335.

[24] Y. Kang, X. Yang, L. Zhang, X. Luo, Y. Xu, H. Wang, J. Liu, MGMFN: multi-graph and MLP-mixer fusion network for Chinese social network sentiment classification, Multimed. Tools Appl. (2024) 1–22.

[25] V.S. Anoop, C.S. Krishna, U.H. Govindarajan, Graph embedding approaches for social media sentiment analysis with model explanation, Int. J. Inf. Manage. Data Insights 4 (1) (2024) 100221.

[26] R. Olusegun, T. Oladunni, H. Audu, Y.A.O. Houkpati, S. Bengesi, Text mining and emotion classification on monkeypox Twitter dataset: a deep learning-natural language processing (NLP) approach, IEEe Access. 11 (2023) 49882–49894.

[27] L. Yang, H. Chen, Z. Li, X. Ding, X. Wu, Give us the facts: enhancing large language models with knowledge graphs for fact-aware language modeling, IEEE Trans. Knowl. Data Eng. (2024).

[28] Chen, Z., Zhang, Y., Fang, Y., Geng, Y., Guo, L., Chen, X., Li, Q., Zhang, W., Chen, J., Zhu, Y. and Li, J., 2024. Knowledge graphs meet multi-modal learning: a comprehensive survey. arXiv preprint arXiv:2402.05391.

[29] A. Bennett, N. Kallus, Proximal reinforcement learning: efficient off-policy evaluation in partially observed markov decision processes, Oper Res 72 (3) (2024) 1071–1086.

[30] P. Adjei, N. Tasfi, S. Gomez-Rosero, M.A. Capretz, Safe reinforcement learning for arm manipulation with constrained Markov decision process, Robotics 13 (4) (2024) 63.

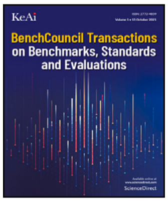
[31] N. Thakur, Monkeypox2022tweets: The fIrst Public Twitter dAtaset On the 2022 MonkeyPox oUtbreak, 2022 2022060172, https://doi.org/10.20944/preprints202206.0172.v1. Preprints.

[32] Gaurav Meena, Krishna Kumar Mohbey, Sunil Kumar, K. Lokesh, A hybrid deep learning approach for detecting sentiment polarities and knowledge graph representation on monkeypox tweets, Decision Anal. J. 7 (2023) 100243, https://doi.org/10.1016/j.dajour.2023.100243. ISSN 2772-6622.

Contents lists available at ScienceDirect

BenchCouncil Transactions on Benchmarks, Standards and Evaluations

journal homepage: www.keaipublishing.com/en/journals/benchcouncil-transactions-on-benchmarks-standards-and-evaluations/


KeAi
BenchCouncil Transactions on Benchmarks, Standards and Evaluations

Full length article

Patrick Star: A comprehensive benchmark for multi-modal image editing

Di Cheng [a], ZhengXin Yang [b], ChunJie Luo [b], Chen Zheng [c,d,e], YingJie Shi [a,*]

[a] *School of Arts & Sciences, Beijing Institute of Fashion Technology, Beijing, China*
[b] *Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China*
[c] *Institute of Software, Chinese Academy of Sciences, Beijing, China*
[d] *University of Chinese Academy of Sciences, Nanjing, China*
[e] *Hangzhou Institute for Advanced Study, University of Chinese Academy of Sciences, Hangzhou, China*

ARTICLE INFO

Keywords:
Image editing
AIGC
Benchmark

ABSTRACT

Generative image editing enhances and automates traditional image designing methods. However, there is a significant imbalance in existing research, where the development of sketch-guided and example-guided image editing has not been sufficiently explored compared to text-guided image editing, despite the former being equally important in real-world applications. The leading cause of this phenomenon is the severe lack of corresponding benchmark datasets. To address this issue, this paper proposes a comprehensive and unified benchmark dataset, Patrick Star, which consists of approximately 500 test images, to promote balanced development in this field across multi-task and multi-modal settings. First, theoretical analysis grounded in Evaluatology highlights the importance of establishing a balanced benchmark dataset to advance research in image editing. Building on this theoretical foundation, the dataset's construction methodology is explained in detail, ensuring it addresses critical gaps in existing studies. Next, statistical analyses are conducted to verify the dataset's usability and diversity. Finally, comparative experiments underscore the dataset's potential as a comprehensive benchmark, demonstrating its capacity to support balanced development in image editing.

1. Introduction

Image editing has emerged as a crucial direction in both industry and academia, particularly as digital content creation becomes increasingly central to modern communication, entertainment, and business operations. Modern generative image editing approaches leverage deep learning models that use conditional information as guidance to achieve intelligent image manipulation. These approaches overcome traditional limitations not only by reducing editing time and improving efficiency, but also by lowering the technical barriers for users. Additionally, these AI-powered editing tools have revolutionized the creative workflow by enabling more intuitive and precise control over image modifications, marking a significant departure from conventional pixel-level manipulation methods. The field of image editing has attracted increasing research attention, evolving from single-modal to multi-modal approaches. Various image editing tasks have been developed, including but not limited to text-guided image editing, sketch-guided image editing, and example-based image editing. Each modality offers unique advantages: text guidance provides natural language interaction, sketch guidance enables precise spatial control, and example-guided approaches allow for intuitive style and content transfer. The integration of these different modalities has opened new possibilities for more flexible and powerful image editing systems.

Recent research [1] reveals an imbalance in the development of these three image-editing approaches. The emergence of CLIP [2] sparked significant advances in text–image alignment, leading to a boom in text-guided image editing research. Further more, the widespread adoption of text prompts in commercial applications, due to their user-friendly interaction mode, has inadvertently led to relatively less attention being paid to sketch-guided and example-guided editing approaches. However, alternative guidance methods are equally important as guided approaches in the field of image editing, particularly in scenarios where precise visual control or style matching is crucial. Sketch-guided editing, for instance, offers invaluable advantages in professional design workflows where exact spatial arrangements are required, while example-guided methods excel in maintaining visual consistency and achieving complex shape transfers that may be difficult to describe through text alone.

A fundamental shift in research paradigms lies at the root of this imbalance. As the field evolves from image generation to image editing, the nature of sketch-guided image editing tasks has transcended traditional image translation. However, this transformation appears to

* Corresponding author.
E-mail address: shiyingjie1983@163.com (Y. Shi).

https://doi.org/10.1016/j.tbench.2025.100201
Received 13 February 2025; Received in revised form 29 March 2025; Accepted 8 April 2025
Available online 30 April 2025

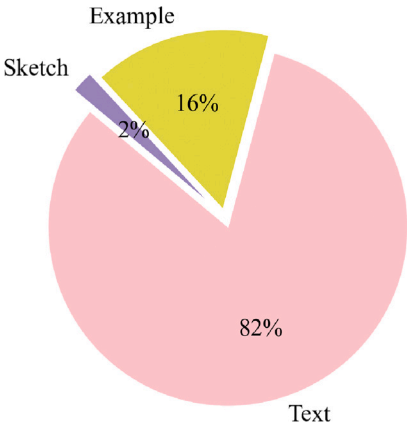


Fig. 1. The distribution of image editing tasks is imbalanced.

have been overlooked by many researchers who may perceive the field as exhausted, failing to recognize the new possibilities and challenges that emerge in the context of editing rather than generation. Meanwhile, example-guided image editing remains at a nascent stage, with current approaches primarily limited to surface-level manipulations and complete object transfers. These methods have yet to achieve the granular control and sophisticated content manipulation capabilities that modern image editing demands. The relative immaturity of these approaches stems from their current inability to handle partial object modifications or more nuanced content transformations. This fundamental misalignment in research focus and the early developmental stage of example-guided methods have contributed to a significant disparity in benchmark datasets compared to text-guided image editing tasks. The lack of comprehensive benchmarks is not merely a data collection issue, but rather a reflection of the deeper challenges in understanding and defining the full potential of these guidance modalities in the context of modern image editing.

Based on our literature review [3], as shown in Fig. 1, current research in the field of image editing exhibits a significant imbalance across different guidance modalities. This uneven distribution of research attention has led to two critical challenges: First, when conducting comparative experiments, niche tasks such as sketch-guided and example-guided editing struggle to find appropriate control groups. Researchers often resort to direct comparisons with text-guided methods, a practice that presents methodological limitations. Second, existing benchmark datasets suffer from two major deficiencies:

- Fragmentation of Test Content: Test data for different guidance modalities (text, sketch, and example) are typically isolated, with almost no benchmark datasets supporting evaluation across all three editing modes simultaneously. This fragmentation makes cross-modal performance comparisons both challenging and less convincing.
- Inconsistency in Evaluation Metrics: Taking Pre_error (used to evaluate the preservation of non-edited regions) as an example, the accuracy of such evaluation metrics heavily depends on the uniformity of editing region sizes. However, existing benchmark datasets often overlook this crucial factor by not standardizing the editing regions, directly impacting the comparability and reliability of evaluation results.

We present Patrick Star Bench, a benchmark dataset designed specifically for image editing tasks. Addressing the issues of inconsistent evaluation standards and fragmented test content in existing datasets, we developed a systematic benchmark construction method from the perspective of scientific evaluation. as illustrated in Fig. 2. This comprehensive dataset consists of five components:Source Image, Mask, Prompt, Sketch, Example and GroundTruth. Patrick Star encompasses 500 sets totaling 2,500 images, covering three major tasks and seven subtasks. The main characteristics and contributions of this benchmark dataset are as follows: First, it achieves unified support for three major editing modalities text-guided, sketch-guided, and example-guided editing, providing a reliable standard for evaluating model performance across cross-modal editing tasks. The dataset implements a rigorous quality control system, ensuring alignment between prompts and images, precision of mask boundaries, and clarity of line extractions. Through experimental validation on six representative models, Patrick Star Bench demonstrates strong discriminative power and reliability. The experimental results indicate that this benchmark not only effectively evaluates performance differences across various editing methods but also provides valuable reference points for subsequent model improvements. we developed a systematic benchmark construction method from the perspective of scientific evaluation. The main contributions are:

(1) Cross-Modal Integration: For the first time, a benchmark dataset unifies the evaluation of text-guided, sketch-guided, and example-guided editing within a single framework. This directly addresses the fragmentation challenge in existing benchmarks and enables reliable cross-modal performance comparisons.

(2) Standardized Evaluation Framework: By implementing consistent mask regions and evaluation metrics across all modalities, our benchmark resolves the long-standing issue of inconsistent evaluation standards, particularly in measuring preservation errors (Pre_error) across different editing approaches.

(3) Extensive Validation: Through rigorous experiments with six representative models, Patrick Star Bench demonstrates strong discriminative power in identifying performance differences across various editing methods.

These contributions collectively address the key challenges in current image editing evaluation and establish a more robust and comprehensive evaluation standard for the field.

2. Related work

2.1. Image editing methods

Image editing has evolved from single-modal approaches to multimodal methodologies, encompassing text-guided, sketch-guided, and example-guided editing techniques. Among these, text-guided image editing has experienced remarkable growth with numerous downstream applications, including instructional editing, position modification [4,5], object manipulation [6–9] (movement, deletion, and addition), and scene reconstruction. These methods typically leverage pre-trained models through fine-tuning or task-specific adapters, achieving impressive results while reducing computational costs. The technical paradigm in image editing has shifted from GANs to Diffusion Models, with each advancement demanding larger datasets and more parameters. However, this trend toward increasingly resource-intensive models poses challenges for general users who may lack access to sufficient computational resources. Sketch-guided image editing has undergone a significant transformation. In its early stages, the field primarily focused on direct image-to-image translation within sketched regions. However, contemporary approaches have evolved to utilize sketches as auxiliary conditional controls for content generation in specific domains. Despite this advancement, the application of sketch guidance in local image editing tasks remains relatively unexplored, representing a notable gap in current research. Example-guided image editing currently encompasses two primary approaches. The first method focuses on object transfer through segmentation, which preserves the complete set of object characteristics but often struggles with seamless integration, particularly when dealing with non-independent objects. The second approach leverages semantic information extracted from reference images to generate semantically consistent results. While this method excels at contextual integration, it may not fully

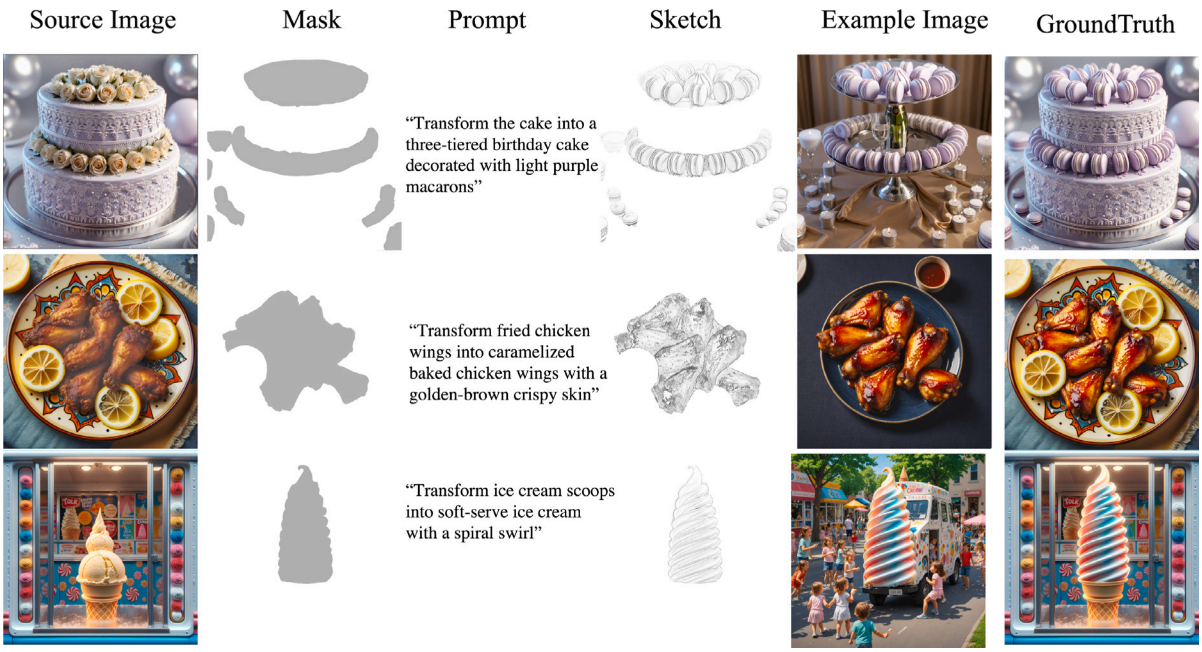

Source Image
Mask
Prompt
Sketch
Example Image
GroundTruth
“Transform the cake into a three-tiered birthday cake decorated with light purple macarons”
“Transform fried chicken wings into caramelized baked chicken wings with a golden-brown crispy skin”
“Transform ice cream scoops into soft-serve ice cream with a spiral swirl”

Fig. 2. Patrick Star: Cases for Image editing tests.

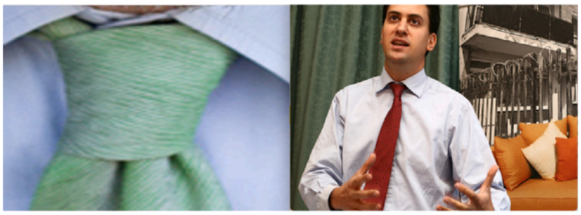


Fig. 3. COCOEE pair of test cases.

preserve specific object details. This creates a fundamental trade-off between feature preservation and contextual harmony, highlighting the challenge of balancing object fidelity with seamless scene integration in example-guided editing.

2.2. Image editing evaluation benchmarks

While numerous evaluation benchmarks exist in the image editing field, these benchmark datasets commonly exhibit significant limitations. As shown in Table 1, existing benchmarks typically support only a single guidance modality: text guidance (e.g., EditBench [10], TedBench [5]), example guidance (e.g., COCOEE [11]), or sketch guidance (e.g., SKETCH Dataset [12]). Through in-depth analysis of existing benchmarks, we identify several key challenges:

First, the limitation of evaluation paradigms. EditBench [10] primarily focuses on text and mask-guided inpainting while neglecting global editing tasks; TedBench [5], despite expanding the task scope, lacks detailed instructions; EditVal [13] is constrained by the low resolution and blurry image quality inherited from the MS-COCO dataset [14] and Emu Edit relies solely on input images from the MagicBrush [15] benchmark. Such singular evaluation perspectives fail to comprehensively reflect model performance.

Second, the absence of cross-modal support. Although COCOEE [11] attempts to support multiple guidance modalities through data processing, its scope remains limited to simple editing tasks within object detection contexts. As illustrated in Fig. 3, this dataset exhibits inconsistencies between reference images and ground truth, highlighting the technical challenges in constructing high-quality multi-modal editing benchmarks: maintaining complex non-independent editing elements (such as modifying a round collar to a notched lapel) while ensuring content and style consistency between target and groundtruth images.

To address these challenges, we propose Patrick Star Bench with a more systematic task classification system. For simple tasks, it includes quantity changes color modifications, position adjustments, and basic state transformations, focusing on evaluating models’ local precise editing capabilities. For complex tasks, it encompasses material transformations, content synthesis, overall consistency, and texture transformations, comprehensively testing models’ ability to handle sophisticated editing scenarios. This classification not only covers the seven types of editing operations in traditional benchmarks (background modification, global transformation, style transfer, object removal, addition, local editing, and texture/color changes) but also provides more fine-grained evaluation criteria.

More importantly, Patrick Star Bench pioneers unified support for text, example, and sketch guidance, establishing a more comprehensive and reliable standard for evaluating cross-modal image editing capabilities. Through systematic data construction processes and strict quality control, we have successfully addressed the challenges of data consistency and evaluation standard uniformity.

3. Dataset construction

3.1. Semantic tag specification

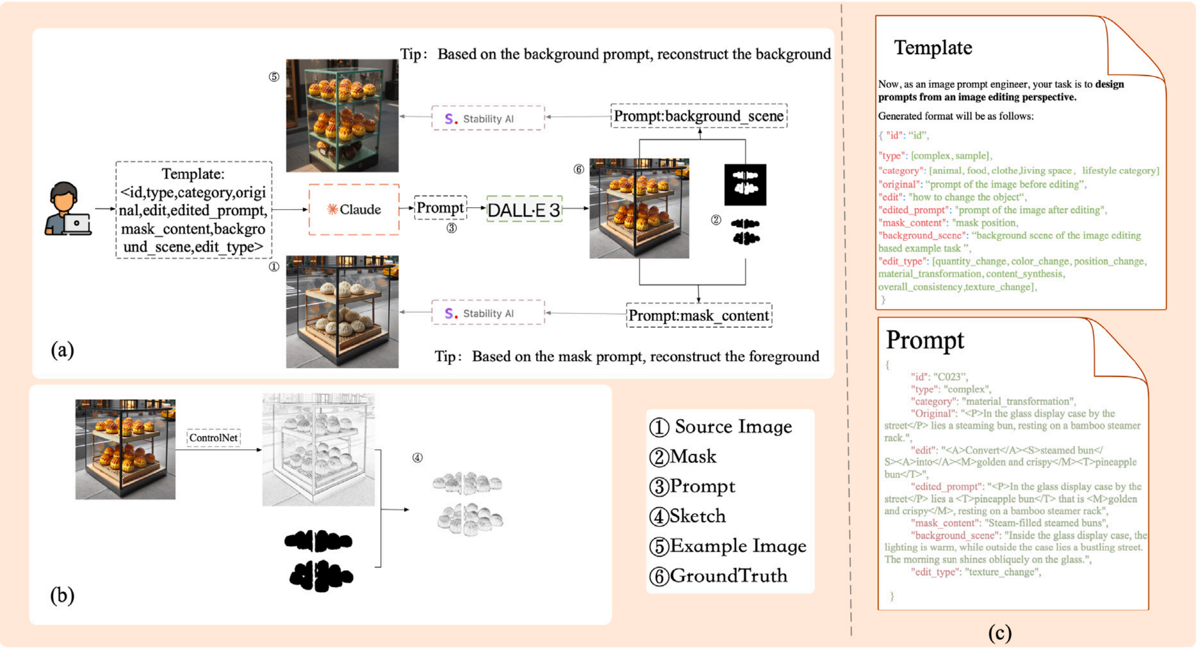
To standardize prompt generation and validation processes in image editing tasks, we designed a strict semantic tag specification. As show in Fig. 7, this specification consists of six fundamental semantic tags: Position tags <P>, Object tags <O>, State tags <S>, Material tags <M>, Action tags <A>, and Temporal tags <T>. These tags are embedded during prompt generation, ensuring that editing requirements have clear structural characteristics.

This tag specification offers the following advantages:

- **Automated Analysis:** Through clear tag boundaries, different types of editing operations can be programmatically extracted and analyzed. For example, we can quickly analyze the distribution of different object categories (via <O> tags) in the dataset, or assess the success rate of specific material transformations (via <M> tags).

Table 1
Comparison of benchmark dataset characteristics. The number of supported types indicates how many guidance types (text/sketch/example) the dataset supports. Patrick Star Bench is the only dataset that supports all three types of guidance.

Dataset	Source image	Text	Sketch	Example image	Mask	Groundtruth	Number of supported types
Text-guided							
EditBench [10]	✓	✓			✓	✓	1
TedBench [5]	✓	✓				✓	1
EditVal [13]	✓	✓				✓	1
IP2P [16]	✓	✓				✓	1
MagicBrush [15]	✓	✓			✓	✓	1
Emu Edit [9]	✓	✓					1
Example-guided							
COCOEE [11]	✓			✓	✓	✓	1
Sketch-guided							
SKETCH Dataset [12]			✓			✓	1
Patrick Star Bench	✓	✓	✓	✓	✓	✓	**3**


Tip: Based on the background prompt, reconstruct the background
⑤
Stability AI
Prompt:background_scene
Template:
<id,type,category,original,edit,edited_prompt,
mask_content,background_scene,edit_type>
Claude
Prompt
③
DALL·E 3
⑥
②
①
Stability AI
Prompt:mask_content
Tip: Based on the mask prompt, reconstruct the foreground
(a)
ControlNet
④
(b)
① Source Image
②Mask
③Prompt
④Sketch
⑤Example Image
⑥GroundTruth
(c)

Template

Now, as an image prompt engineer, your task is to **design prompts from an image editing perspective.**

Generated format will be as follows:

```
{ "id": “id”,
"type": [complex, sample],
"category": [animal, food, clothe,living space ,  lifestyle category]
"original": “prompt of the image before editing”,
"edit": "how to change the object“,
"edited_prompt": "prompt of the image after editing",
"mask_content": "mask position,
"background_scene": “background scene of the image editing
based example task ”,
"edit_type": [quantity_change, color_change, position_change,
material_transformation, content_synthesis,
overall_consistency,texture_change],
}
```

Prompt

```
{
        "id": "C023”,
        "type": "complex",
        "category": "material_transformation",
        "Original": "<P>In the glass display case by the
street</P> lies a steaming bun, resting on a bamboo steamer
rack.",
        "edit": "<A>Convert</A><S>steamed bun</
S><A>into</A><M>golden and crispy</M><T>pineapple
bun</T>",
        "edited_prompt": "<P>In the glass display case by the
street</P> lies a <T>pineapple bun</T> that is <M>golden
and crispy</M>, resting on a bamboo steamer rack",
        "mask_content": "Steam-filled steamed buns",
        "background_scene": "Inside the glass display case, the
lighting is warm, while outside the case lies a bustling street.
The morning sun shines obliquely on the glass.",
        "edit_type": "texture_change",

 }
```

Fig. 4. Overview of Patrick Star dataset construction pipeline. (a) illustrates our multi-stage generation process: We first use Claude to generate structured prompts following our template format, then employ DALL-E 3 to create the ground truth image. After manual mask annotation or SAM [17] annotation on the ground truth, we utilize Stable Diffusion inpainting in two ways: combining the ground truth with mask and background_scene prompt to generate the source image (pre-editing state), and combining the ground truth with inverted mask and mask_content prompt to create the example image. (b) shows the sketch generation process, where we apply ControlNet's preprocessor to extract structural features from the source image and combine them with the mask to obtain the final line drawing. (c) presents our template structure for prompts, which includes comprehensive fields for image metadata, editing specifications, and contextual information. The complete dataset comprises seven essential components: source image, mask, text prompt, sketch, example image, ground truth, and mask content, collectively forming a comprehensive multi-modal benchmark for image editing evaluation.

- **Consistency Verification:** Using tag correspondence, we can automatically verify semantic consistency between pre- and post-editing descriptions. Particularly in complex editing tasks, these tags help track changes in key attributes.
- **Quality Control:** By enforcing tag specifications during the generation phase, we significantly reduce the need for manual review later in the process, improving the efficiency of dataset construction.

This tag specification serves as a crucial foundation for building Patrick Star Bench, providing reliable support for subsequent automated processing and analysis.

3.2. Multi-dimensional task taxonomy

To comprehensively evaluate image editing models' performance, we propose a two-level task classification system. Based on the complexity of editing operations and semantic levels, Patrick Star Bench categorizes tasks into Simple Tasks and Complex Tasks.

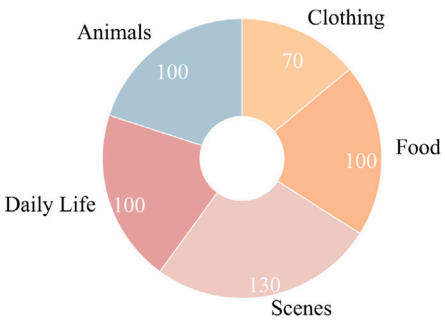


Fig. 5. Image category quantity chart.

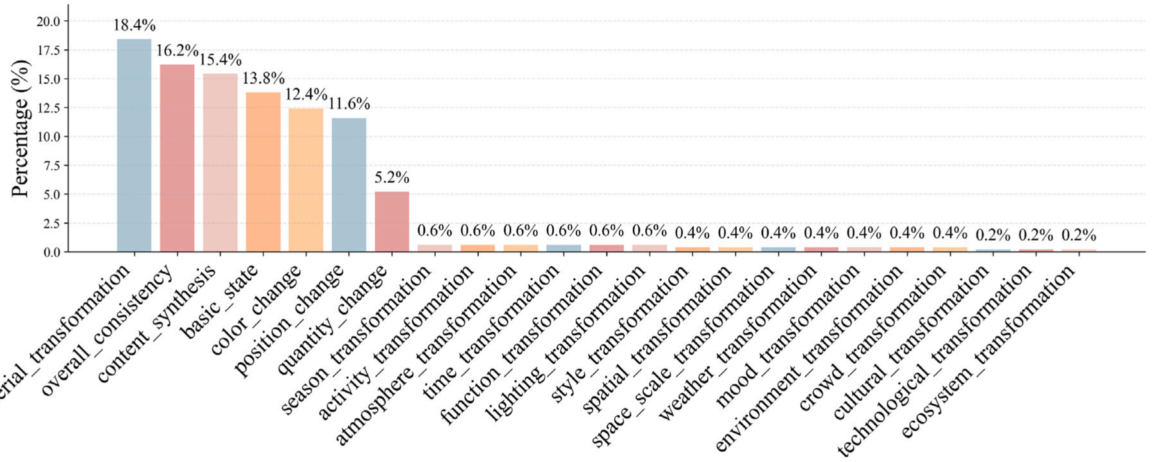


Fig. 6. Distribution of image editing operations in our dataset.

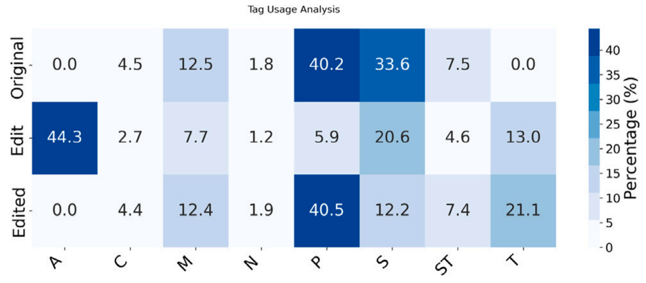


Fig. 7. A heatmap of the label and the changes in the quantities before, during, and after editing.

At the basic level, we define four fundamental editing operations: `quantity_change` focuses on precise modifications of object numbers in scenes; `color_change` addresses adjustments to basic appearance attributes; `position_change` evaluates models' capabilities in spatial layout modifications; and `basic_state` tests simple object attribute transformations. While these tasks are operationally simple, they require models to possess precise local editing capabilities.

Complex tasks examine models' ability to handle complex semantic transformations: `material_transformation` requires changing object physical properties while maintaining shape; `content_synthesis` tests models' ability to integrate multiple elements; `overall_consistency` evaluates scene coherence during large-scale editing; and `texture_ transformation` focuses on fine-grained surface feature modifications. These tasks not only demand accurate editing operations but also require maintaining natural contextual transitions. To be specific, the taxonomy categorizes the distribution of image editing operations, as shown in Fig. 6, while more illustrative examples of image editing are presented in Fig. 8.

3.3. Dataset construction pipeline

Our dataset construction approach combines both real-world photography and AI-generated content to ensure diversity and quality, as shown in Fig. 4.Specifically, our dataset consists of 100 high-quality images sourced from Unsplash, a copyright-free photography platform, and 400 AI-generated images. This hybrid approach leverages the authenticity of real photographs while maintaining scalability through generative models.

3.3.1. Dual-source data collection

For real-world images, we carefully selected 100 high-resolution photographs from Unsplash that serve as ground truth images. These images were processed through multimodal large language models to automatically generate initial descriptions following our template format, followed by manual refinement to ensure tag specification compliance. Masks for these images were generated using a hybrid approach: for regions identifiable by SAM, we automatically expanded the bounding boxes and applied masks accordingly, while intricate details were manually annotated by experts to ensure precise editing region definition. The corresponding source images and reference images were then created using Stable Diffusion inpainting models with varying prompts, maintaining consistency with our editing objectives.

For the AI-generated portion, we develop a systematic creation process starting with prompt design. The process begins with designing specific prompt templates for each task type that comply with our tag specification while capturing the core challenges of each editing operation. For example, in material transformation tasks, the template must explicitly specify the material characteristics before and after editing while maintaining other object attributes unchanged.

3.3.2. Content creation

The content creation phase employs different strategies based on the image source. For Claude AI-generated [18] content, we utilize the DALL-E3 [19] API with optimized parameters, requiring multiple generation attempts to obtain candidates that best align with the prompts. For Unsplash-sourced images, we employ Stable Diffusion inpainting to generate variations while preserving the high quality of the original photographs. In both cases, mask generation focuses on precise editing region definition, and sketch extraction utilizes ControlNet with optimized parameters.

3.3.3. Quality verification

Our quality verification process ensures consistency across both real and generated images. For Unsplash-sourced content, we pay particular attention to the quality of generated variations and their alignment with the original photographs. For AI-generated content, we focus on the consistency between different versions of the same scene. The automated system verifies tag consistency, cross-modal alignment, and editing region standardization, while task-specific verification ensures that each sample meets its unique requirements.

Through this hybrid approach, we created a dataset of 500 high-quality editing samples, combining the authenticity of real photographs with the scalability of AI generation. This combination provides a more comprehensive benchmark for evaluating image editing models, as it tests performance on both real-world photographs and AI-generated content. The dataset's diverse sources and standardized quality make it particularly valuable for assessing models' generalization capabilities across different image types and editing scenarios.

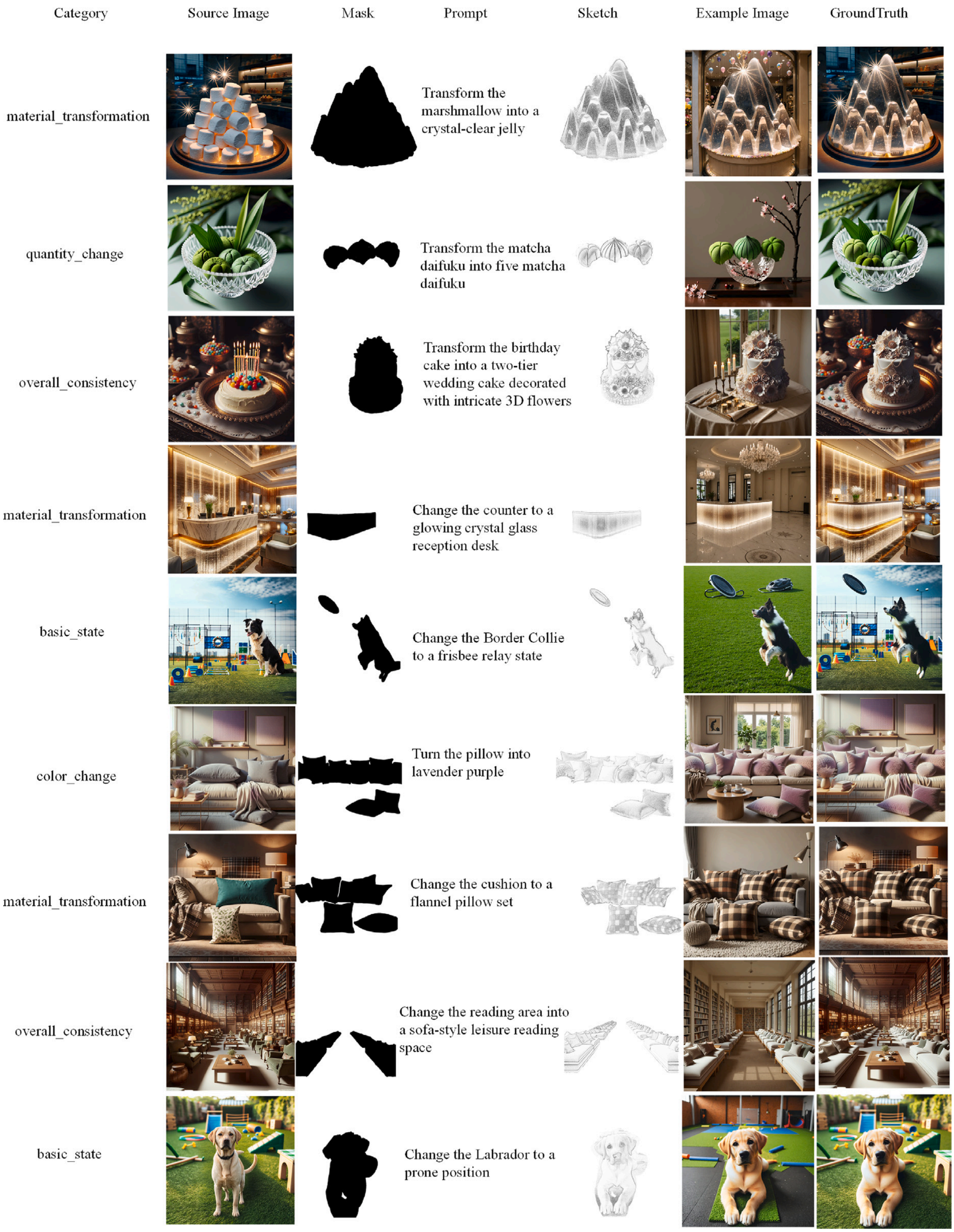

Category
Source Image
Mask
Prompt
Sketch
Example Image
GroundTruth
material_transformation
Transform the marshmallow into a crystal-clear jelly
quantity_change
Transform the matcha daifuku into five matcha daifuku
overall_consistency
Transform the birthday cake into a two-tier wedding cake decorated with intricate 3D flowers
material_transformation
Change the counter to a glowing crystal glass reception desk
basic_state
Change the Border Collie to a frisbee relay state
color_change
Turn the pillow into lavender purple
material_transformation
Change the cushion to a flannel pillow set
overall_consistency
Change the reading area into a sofa-style leisure reading space
basic_state
Change the Labrador to a prone position

Fig. 8. Illustration of five categories of food image editing: material transformation, quantity change, overall consistency, basic state and color change. Each row demonstrates a specific editing type with its source image, binary mask, text prompt, generated sketch, example reference, and ground truth (GT) result.

Table 2
Patrick Star Bench's test results on different tasks.

Evaluation metric	Text-guided		Sketch-guided		Example-guided	
	Image inpainting-SD1.5	Image inpainting-SDXL [20]	Controlnet-SD2.1	Controlnet-SDXL [4]	Paint by example [11]	DesignEdit [21]
LPIPS↓	0.0978	**0.0678**	0.572	**0.473**	0.0948	**0.0284**
FID↓	22.270	**19.314**	45.156	**34.253**	22.785	**18.501**
Pre_error↓	0.126	**0.103**	–	–	0.126	**0.099**
CLIP_Score↑	65.9120	**71.2986**	–	–	**70.7843**	70.1917
SSIM ↑	0.8214	**0.8442**	0.3282	**0.4885**	0.8253	**0.8748**
Aesthetic Score ↑	**4.9033**	4.8567	**5.0463**	4.8899	4.9635	**5.1798**

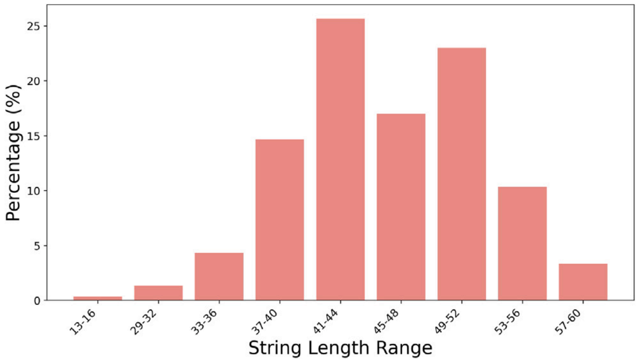


Fig. 9. Instruction string length.

3.4. Dataset statistics

To ensure comprehensive coverage and balance of the dataset, we conducted a detailed statistical analysis of various dataset features. As shown in Fig. 9, the instruction lengths exhibit diversity, with some being short for simple tasks like object replacement and color changes, while others are longer and more descriptive. These longer instructions typically require a certain level of detail to guide the model in making precise adjustments.

The source images in our dataset span across five main categories, as illustrated in Fig. 5: Natural Elements (25.0%), Daily Item Surfaces (23.0%), Clothing Parts (21.3%), Household Items (18.3%), and Food & Beverages (12.4%). This balanced distribution ensures the dataset's representativeness across different domains and editing scenarios. The relatively higher proportions of natural elements and surface textures reflect common editing requirements in real-world applications, while the inclusion of diverse categories enables comprehensive evaluation of models' generalization capabilities across different editing tasks.

4. Experiments

We conducted three groups of image editing experiments, totaling six tests covering both basic and optimized versions. All experiments were performed on an RTX 3090 GPU with 24 GB memory using identical hyperparameters, ensuring fair comparison conditions. For text-guided image editing, we employed SD1.5-Inpainting and SDXL-Inpainting models. Given the relatively limited work in sketch-guided image editing, we opted to use the ControlNet sketch generation models in both SD2.1 and SDXL versions. For example-guided image editing tasks, we compared Paint by Example with our proposed DesignEdit method. We evaluated the models using six evaluation metrics: LPIPS, FID and SSIM [22] for image quality assessment, Aesthetic Score [23] for image aesthetic evaluation, and CLIP_Score [24] and Pre_error for image content assessment.

The experimental results reveal significant performance variations across different approaches. As shown in Table 2, SDXL consistently outperforms SD1.5 and SD2.0 across most evaluation metrics, including LPIPS, FID, CLIP Score, and SSIM. This demonstrates that our dataset effectively differentiates the generation capabilities of different models. The ability to highlight these variations confirms the dataset's robustness in benchmarking image editing performance across multiple tasks and model architectures.

Human Evaluation We selected images generated by the six methods mentioned above and presented them to 100 participants. Participants were allowed to select multiple images that met the specified criteria. As shown in Fig. 10, they were asked to evaluate the images based on realism and alignment with the provided instructions. Inpainting-sd1.5, Inpainting-sdxl, Paint by Example, and DesignEdit received high scores for both realism and alignment. In contrast, ControlNet sdxl and ControlNet sd1.5 had lower scores, which demonstrates that our benchmark can effectively distinguish output images quality. Furthermore, the smaller differences in the ControlNet methods indicate that the benchmark can also capture subtle variations in performance. Overall, these results prove that the benchmark is effective and sensitive, providing valuable guidance for image editing tasks.

These comprehensive experimental results not only verify our dataset's applicability across different guidance modes but also demonstrate its effectiveness in evaluating and differentiating the performance of various methods. The consistent performance improvements observed in the optimized versions further confirm our dataset's discriminative capability and reliability, establishing a dependable benchmark for future model improvements and evaluations.

5. Conclusion and future work

This paper presents Patrick Star Bench, a comprehensive evaluation benchmark designed specifically for image editing tasks. Addressing the challenges of inconsistent evaluation standards and fragmented testing content in existing datasets, we have developed a systematic benchmark construction methodology grounded in scientific evaluation principles.

The key features and contributions of our benchmark dataset are significant. Notably, it is the first to provide unified support for three major editing paradigms: text-guided, sketch-guided, and example-guided editing. This integration establishes a reliable standard for evaluating model performance across cross-modal editing tasks. The dataset implements a rigorous quality control system that ensures prompt-image alignment, mask boundary precision, and sketch extraction clarity.

Through extensive validation experiments across six representative models, Patrick Star Bench has demonstrated excellent discriminative capability and reliability. The experimental results confirm that our benchmark can effectively assess performance differences between various editing methods while providing a solid foundation for future model improvements.

Looking ahead, we envision several promising directions for future research:

(1) Multi-turn Interactive Editing: Extending the benchmark to support evaluation of conversational image editing systems, where multiple rounds of user feedback and model responses are involved.

(2) Dynamic Assessment Metrics: Developing more sophisticated evaluation metrics that can capture the nuanced aspects of interactive editing processes.

(3) Temporal Consistency: Incorporating evaluation criteria for video editing tasks and sequential image modifications.

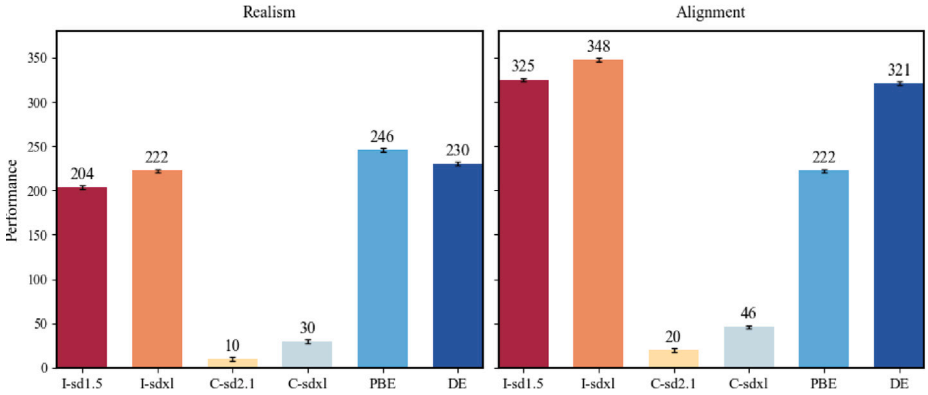


Fig. 10. Human evaluation of image realism and text–image alignment on Patrick Star.

These future developments aim to enhance the benchmark's functionality and broaden its applications in multimodal image editing research. We hope the foundation laid by Patrick Star Bench can contribute to the development of better evaluation methods for image editing technologies.

CRediT authorship contribution statement

Di Cheng: Writing – original draft, Conceptualization. **ZhengXin Yang:** Formal analysis, Conceptualization. **ChunJie Luo:** Methodology, Conceptualization. **Chen Zheng:** Data curation, Conceptualization. **YingJie Shi:** Writing – review & editing.

Declaration of competing interest

The author Chunjie Luo is the Assistant Editor in Chief and Chen Zheng is the Associate Editor for the journal BenchCouncil Transactions on Benchmarks, Standards and Evaluations and were not involved in the editorial review or the decision to publish this article. The other authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We would like to express our gratitude to the reviewers and editors for their constructive comments and suggestions. We also sincerely thank the teachers and students from the Institute of Computing Technology, Chinese Academy of Sciences, for their valuable feedback and support.

References

[1] Y. Huang, J. Huang, Y. Liu, M. Yan, J. Lv, J. Liu, W. Xiong, H. Zhang, S. Chen, L. Cao, Diffusion model-based image editing: A survey, 2024, arXiv:2402.17525.

[2] A. Radford, J.W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., Learning transferable visual models from natural language supervision, in: International Conference on Machine Learning, PMLR, 2021, pp. 8748–8763.

[3] S. Basu, M. Saberi, S. Bhardwaj, A.M. Chegini, D. Massiceti, M. Sanjabi, S.X. Hu, S. Feizi, EditVal: Benchmarking diffusion based text-guided image editing methods, 2023, arXiv:2310.02426.

[4] P. Li, Q. Huang, Y. Ding, Z. Li, LayerDiffusion: Layered controlled image editing with diffusion models, 2023, arXiv:2305.18676.

[5] B. Kawar, S. Zada, O. Lang, O. Tov, H. Chang, T. Dekel, I. Mosseri, M. Irani, Imagic: Text-based real image editing with diffusion models, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 6007–6017.

[6] J. Zhuang, Y. Zeng, W. Liu, C. Yuan, K. Chen, A task is worth one word: Learning with task prompts for high-quality versatile image inpainting, in: European Conference on Computer Vision, Springer, 2025, pp. 195–211.

[7] Z. Geng, B. Yang, T. Hang, C. Li, S. Gu, T. Zhang, J. Bao, Z. Zhang, H. Li, H. Hu, et al., Instructdiffusion: A generalist modeling interface for vision tasks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 12709–12720.

[8] S. Yang, L. Zhang, L. Ma, Y. Liu, J. Fu, Y. He, Magicremover: Tuning-free text-guided image inpainting with diffusion models, 2023, arXiv preprint arXiv:2310.02848.

[9] S. Sheynin, A. Polyak, U. Singer, Y. Kirstain, A. Zohar, O. Ashual, D. Parikh, Y. Taigman, Emu edit: Precise image editing via recognition and generation tasks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 8871–8879.

[10] S. Wang, C. Saharia, C. Montgomery, J. Pont-Tuset, S. Noy, S. Pellegrini, Y. Onoe, S. Laszlo, D.J. Fleet, R. Soricut, et al., Imagen editor and editbench: Advancing and evaluating text-guided image inpainting, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 18359–18369.

[11] B. Yang, S. Gu, B. Zhang, T. Zhang, X. Chen, X. Sun, D. Chen, F. Wen, Paint by example: Exemplar-based image editing with diffusion models, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 18381–18391.

[12] M. Eitz, J. Hays, M. Alexa, How do humans sketch objects? ACM Trans. Graph. 31 (4) (2012) 1–10.

[13] S. Basu, M. Saberi, S. Bhardwaj, A.M. Chegini, D. Massiceti, M. Sanjabi, S.X. Hu, S. Feizi, Editval: Benchmarking diffusion based text-guided image editing methods, 2023, arXiv preprint arXiv:2310.02426.

[14] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C.L. Zitnick, P. Dollár, Microsoft COCO: Common objects in context, 2015, arXiv:1405.0312.

[15] K. Zhang, L. Mo, W. Chen, H. Sun, Y. Su, Magicbrush: A manually annotated dataset for instruction-guided image editing, Adv. Neural Inf. Process. Syst. 36 (2024).

[16] Q. Guo, T. Lin, Focus on your instruction: Fine-grained and multi-instruction image editing by attention modulation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 6986–6996.

[17] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A.C. Berg, W.-Y. Lo, P. Dollár, R. Girshick, Segment anything, 2023, arXiv:2304.02643.

[18] Anthropic, Claude AI, 2025, https://claude.ai. (Accessed 29 January 2025).

[19] L. Zhang, A. Rao, M. Agrawala, Adding conditional control to text-to-image diffusion models, in: IEEE International Conference on Computer Vision, ICCV.

[20] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J.-Y. Zhu, S. Ermon, SDEdit: Guided image synthesis and editing with stochastic differential equations, 2022, arXiv:2108.01073.

[21] Y. Jia, Y. Yuan, A. Cheng, C. Wang, J. Li, H. Jia, S. Zhang, DesignEdit: Multi-layered latent decomposition and fusion for unified & accurate image editing, 2024, arXiv preprint arXiv:2403.14487.

[22] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Trans. Image Process. 13 (4) (2004) 600–612.

[23] C. Schuhmann, R. Beaumont, LAION-AI aesthetic-predictor, 2022, URL https://github.com/LAION-AI/aesthetic-predictor. (Accessed 27 March 2025).

[24] J. Hessel, A. Holtzman, M. Forbes, R.L. Bras, Y. Choi, CLIPScore: A reference-free evaluation metric for image captioning, 2022, arXiv:2104.08718.