Graph ML-Enabled Molecular Design Assistant Using Graph Neural Networks

Saptarshi Ghosh, Uday Sankar De, Mangaldip Dhua, Arnab Subhra Ghosh, Sumit Chaira, and Kishan Singh*

Abstract Molecular graphs support computational screening of assay activity, but reliable deployment requires consistent features and defensible evaluation. This study compares graph convolutional networks (GCN), edge-aware graph isomorphism networks (GINE), and graph attention networks v2 (GATv2) on twelve Tox21 endpoints. Re-evaluation of three saved checkpoints on the recorded 784-compound test set gives macro ROC-AUCs of 0.830, 0.808, and 0.798, respectively. An implementation audit identifies scaffold overlap in the fingerprint-grouped split and a training–serving feature mismatch. Correcting inference restores checkpoint-compatible predictions. Additional evaluation uses 598 compounds from the NIH final-evaluation set after removing overlap with the original dataset and duplicate parent structures. External discrimination and precision–recall metrics are reported alongside matched-partition fingerprint baselines and molecule-level bootstrap intervals. A Flask interface exposes assay scores, structure depictions, and CSV processing. The contribution is a comparative screening implementation and reproducibility audit, rather than a new neural architecture or molecular generator. The results support retrospective assay-activity prioritization; they do not establish human safety, prospective discovery performance, or stability across independent neural training runs.

Saptarshi Ghosh, Uday Sankar De, Mangaldip Dhua, Arnab Subhra Ghosh, Sumit Chaira, and Kishan Singh
Institute of Engineering & Management, Kolkata; School of University of Engineering and Management, Kolkata, West Bengal, India
e-mail: saptarshi.ghosh2022@uem.edu.in; e-mail: udaysankar.de2022@uem.edu.in;
e-mail: mangaldip.dhua2022@uem.edu.in; e-mail: arnabsubhra.ghosh2022@uem.edu.in;
e-mail: sumit.chaira2022@uem.edu.in; e-mail: kishan.singh@uem.edu.in
*Corresponding author: e-mail: kishanee03@gmail.com.
Kishan Singh: ORCID 0009-0001-9943-4313.

Keywords: Graph Neural Networks; Tox21; Molecular Screening; Assay Activity; Reproducibility

1 Introduction

Molecular property models can help prioritize compounds for experimental investigation. In a molecular graph, atoms form vertices and bonds form edges, allowing graph neural networks (GNNs) to learn representations from connectivity. However, a strong benchmark score is useful only when the data partition, graph encoding, and deployed inference agree. Assay activity must also be distinguished from whole-organism toxicity: a positive response concerns a particular biological assay, while a negative response does not establish that a compound is safe.

This work implements the toxicity-screening component of the Molecular Design Assistant named in the accepted paper. Its demonstrated scope is twelve-endpoint Tox21 classification and an interactive prediction interface. Molecular generation, optimization, and experimental validation of newly designed compounds are outside the implemented scope. The contribution consists of (i) a common four-layer GCN/GINE/GATv2 comparison, (ii) an audit linking saved checkpoints, molecular features, and recorded split identifiers, and (iii) additional evaluation against published NIH assay labels with controlled fingerprint baselines. No novel convolution, loss function, or representation is proposed.

2 Related Work

MoleculeNet established public molecular-learning tasks and evaluation conventions, while message-passing models learn graph-level properties from atom and bond representations [1, 2]. GCN uses degree-normalized neighborhood aggregation [3]. GIN studies the expressiveness of sum aggregation; GINE adds edge-dependent messages [4, 5]. GATv2 changes the order of attention operations to permit query-dependent neighbor ranking [6]. These distinctions motivate a controlled implementation comparison, rather than an assumption that greater expressiveness guarantees better prediction.

Fingerprint-based predictors remain necessary comparators. Yang et al. examine learned molecular representations and the consequences of evaluation splits [7]. AttentiveFP learns attention-based molecular representations [8], while MolPROP evaluates language–graph fusion on scaffold-split tasks

[9]. A recent benchmark of pretrained embeddings further emphasizes comparisons with ECFP representations [10]. Their published scores are not pooled into a ranking here because partitions, pretraining, tuning budgets, and data processing differ. Instead, fingerprint baselines are fitted on the same recorded training compounds used by the evaluated checkpoints.

3 Data, Representations, and Models

3.1 Assays and Missing Labels

The original Tox21 CSV contains 7,831 records and twelve binary endpoints: seven nuclear-receptor-related tasks and five stress-response tasks [1, 11]. The task order is NR-AR, NR-AR-LBD, NR-AhR, NR-Aromatase, NR-ER, NR-ER-LBD, NR-PPAR-γ, SR-ARE, SR-ATAD5, SR-HSE, SR-MMP, and SR-p53. Labels denote assay activity, not a single general toxicity class. Missing labels are excluded independently for each task.

The re-evaluation preserves source CSV row identifiers. In the current RDKit/PyG environment, eight source records fail graph parsing and the current dataset loader drops them. Allowing that removal to shift later row identifiers would evaluate the wrong compounds. The recorded test membership is therefore checked against the notebook's saved 784 indices. All 784 test compounds parse successfully in this environment; invalid source records are documented in the released manifest.

3.2 Graph Features and Preprocessing Audit

PyG's SMILES conversion supplies nine categorical atom fields in order: atomic number, chirality, total degree, formal charge, total hydrogen count, radical-electron count, hybridization, aromaticity, and ring membership. Values are vocabulary indices; they must not be replaced by differently ordered raw RDKit values. Each bond becomes two directed edges with three categorical fields: bond type, stereochemistry, and conjugation [13]. "Three edge features" refers to vector length, not three-dimensional geometry. No Cartesian coordinates enter these models.

The notebook attempts dataset-wide z-scoring by assigning new tensors to graphs returned during dataset iteration. These assignments do not persist into the subsequently reconstructed training-loader graphs. Re-evaluation confirms that the checkpoints expect the original categorical features, with

Table 1 Saved-checkpoint architectures. GATv2 has four concatenated attention heads of width 64. Parameter counts include trainable parameters only.

Model	Parameters	Dropout	Activation	Bond attributes
GCN	336,396	0.25	ReLU	Not consumed
GINE	602,672	0.25	ReLU	Used in messages
GATv2	538,380	0.30	ELU	Not consumed

atom inputs clamped to $[-10, 10]$ inside each model. The corrected service consequently uses the same PyG encoder and omits z-scoring. This reproduces the saved results; it does not endorse clamping categorical indices as an optimal representation. Future retraining should use explicit categorical embeddings or a documented encoding and fit any learned preprocessing on training data only.

3.3 Partition Audit

The recorded partition contains 6,264 training, 783 validation, and 784 test records. Molecules are grouped by identical radius-two, 2,048-bit Morgan fingerprints, groups are ordered by size, and the notebook fills the target partitions. Boundary groups may be divided. This is a fingerprint-grouped partition, not a scaffold-disjoint split. Bemis–Murcko scaffolds describe molecular frameworks and cannot be inferred from fingerprint equality [12, 14].

The audit finds 1,942 training, 358 validation, and 368 test scaffold keys, including an empty key for acyclic molecules and an invalid-input key where relevant. Training and test share 171 scaffold keys. Of the 784 test molecules, 585 have a scaffold key present in training, including 149 acyclic molecules; 436 share a nonempty scaffold. The partitions have no duplicate row indices, but row disjointness does not establish chemical independence. Accordingly, the internal results are not described as unseen-scaffold generalization.

3.4 Implemented Architectures

All three models use four message-passing blocks, batch normalization, a 256-dimensional hidden representation, and concatenated global mean and sum pooling. The pooled 512-vector passes through a 256-unit linear layer and a twelve-output linear layer. Sigmoid transforms the final logits into scores. Table 1 records implementation differences rather than treating parameter count or dropout as identical.

For GCN, with self-loops included in $\widetilde{A} = A + I$,

$$H^{(\ell+1)} = \phi\left(\widetilde{D}^{-1/2}\widetilde{A}\widetilde{D}^{-1/2}H^{(\ell)}W^{(\ell)}\right). \tag{1}$$

The implementation calls `GCNConv` without edge weights or bond attributes. For `GINEConv`, a learned edge projection E_ℓ matches the incoming node width:

$$h_i^{(\ell+1)} = \mathrm{MLP}_\ell\left((1+\epsilon_\ell)h_i^{(\ell)} + \sum_{j\in\mathcal{N}(i)} \mathrm{ReLU}(h_j^{(\ell)} + E_\ell e_{ji})\right). \tag{2}$$

Here `train_eps=False`, so $\epsilon_\ell = 0$ is fixed. Each MLP has two linear layers separated by ReLU. The GATv2 attention head computes

$$s_{ij} = a^\top \mathrm{LeakyReLU}(W_L h_i + W_R h_j), \qquad \alpha_{ij} = \frac{\exp(s_{ij})}{\sum_{k\in\mathcal{N}(i)\cup\{i\}}\exp(s_{ik})}, \tag{3}$$

and aggregates $\alpha_{ij}W_R h_j$ over neighbors and the self-loop. The implementation uses `GATv2Conv` without `edge_dim`; bond attributes therefore do not enter attention. Attention weights alone are not validated chemical explanations.

3.5 Recorded Training Procedure

The notebook uses Adam with learning rate 5×10^{-4} and weight decay 10^{-5}, training batches of 32, and evaluation batches of 64. Training runs for at most 50 epochs; validation macro ROC-AUC selects the saved checkpoint. Cosine warm restarts use $T_0 = 10$, $T_{\mathrm{mult}} = 2$, and minimum learning rate 10^{-6}; the notebook also steps a plateau scheduler. Gradient norm is clipped at 5. The supplied notebook code uses an eight-epoch early-stopping counter, replacing the earlier manuscript's inconsistent ten-epoch description.

For observed-label indicator m_{it} and $p_{it} = \sigma(z_{it})$, the implemented masked objective is

$$\mathcal{L} = -\frac{\sum_{i,t} m_{it}\{w_t y_{it}\log p_{it} + (1-y_{it})\log(1-p_{it})\}}{\sum_{i,t} m_{it}}. \tag{4}$$

Missing targets are replaced by zero before loss computation and their contributions are masked. The historical notebook computes $w_t = n_t^-/n_t^+$ over the full dataset, including held-out labels. This label-prevalence leakage is dis-

closed and remains a limitation of the frozen checkpoints. The checkpoints were not retrained for this audit.

4 Evaluation Protocol

4.1 Independent Source Data and Baselines

The NIH Tox21 Challenge provides a separate final-evaluation set of 647 SMILES records and observed assay labels [15]. They are joined by sample identifier. To reduce overlap, an external molecule is removed if its canonical isomeric SMILES or its stereochemistry-free RDKit fragment-parent key occurs anywhere among the original 7,831 records. Duplicate external parent keys are also removed. This retains 598 compounds. The filters affect cohort composition, so the resulting scores are not official challenge leaderboard scores. Structure non-overlap is also weaker than scaffold disjointness and does not imply prospective validation.

Two fixed ECFP baselines use radius-two 2,048-bit Morgan fingerprints: a 300-tree random forest with square-root feature subsampling and balanced class weights, and L_2-regularized logistic regression with $C = 1$, balanced weights, and the liblinear solver. Each endpoint is fitted separately on observed labels from the recorded training partition; invalid training graphs are excluded. Both use seed 229 and no external-set tuning. Neither baseline uses pretraining or additional labeled compounds.

4.2 Metrics and Uncertainty

ROC-AUC and average precision (AP) are computed per task after excluding missing labels, then macro-averaged with equal task weight. AP summarizes the precision–recall curve using the standard stepwise average-precision definition; it is not presented as trapezoidal PR-AUC. Precision, recall, and F1 use an exploratory fixed threshold of 0.5. These class-weighted sigmoid scores are uncalibrated, and the threshold is not a validated operating point.

For confidence intervals, 1,000 paired molecule-level bootstrap replicates resample rows with replacement, preserving within-molecule endpoint associations and missingness. Macro ROC-AUC is retained when every endpoint contains both classes. The 2.5th and 97.5th percentiles form the interval. Model differences use the same resampled rows. These intervals describe finite-test-set uncertainty conditional on the saved models; they do not mea-

Table 2 Recorded 784-compound test set. ROC-AUC is shown for all GNNs; AP, precision (P), recall (R), and F1 are for GCN at threshold 0.5. Missing labels are omitted per task.

Endpoint	GCN	GINE	GATv2	AP	P	R	F1
NR-AR	0.889	0.832	0.801	0.560	0.284	0.622	0.390
NR-AR-LBD	0.871	0.878	0.824	0.639	0.232	0.679	0.345
NR-AhR	0.890	0.876	0.875	0.573	0.414	0.872	0.562
NR-Aromatase	0.846	0.835	0.837	0.234	0.177	0.758	0.287
NR-ER	0.738	0.726	0.724	0.405	0.278	0.651	0.390
NR-ER-LBD	0.816	0.789	0.771	0.327	0.218	0.667	0.328
NR-PPAR-γ	0.786	0.764	0.749	0.220	0.110	0.583	0.185
SR-ARE	0.804	0.765	0.769	0.404	0.366	0.789	0.500
SR-ATAD5	0.830	0.804	0.802	0.214	0.121	0.750	0.209
SR-HSE	0.799	0.755	0.795	0.315	0.182	0.674	0.287
SR-MMP	0.875	0.858	0.852	0.579	0.413	0.871	0.561
SR-p53	0.817	0.811	0.780	0.225	0.143	0.750	0.240

sure training-seed variation, scaffold-cluster dependence, or multiplicity-adjusted significance.

5 Results and Discussion

5.1 Recorded Test Set and Deployment Consistency

Checkpoint-compatible evaluation gives GCN 0.8301, GINE 0.8076, and GATv2 0.7982 macro ROC-AUC. GCN's bootstrap interval is [0.807, 0.854]. Its margins over GINE and GATv2 are approximately 0.0225 and 0.0319 AUC units, respectively, rather than relative percentage gains. Table 2 expands reporting to task-wise ROC-AUC and GCN threshold metrics. GCN has the highest ROC-AUC on eleven endpoints; GINE is highest on NR-AR-LBD. The ranking concerns these checkpoints and this partition, not a general superiority of local chemical interactions over global structure.

The original web app uses reordered raw RDKit atom features, a different bond encoding, and z-scoring that was absent from the effective training graphs. Its macro ROC-AUCs on the same test compounds are only 0.526, 0.560, and 0.535 for GCN, GINE, and GATv2. Correcting only the feature order while retaining z-scoring is insufficient: GCN remains at 0.510. Figure 1 isolates this deployment mismatch. The corrected service uses checkpoint-compatible graph inputs; eighteen predictions spanning six test molecules and all three models agree with batched evaluation within 2×10^{-6} absolute score error.

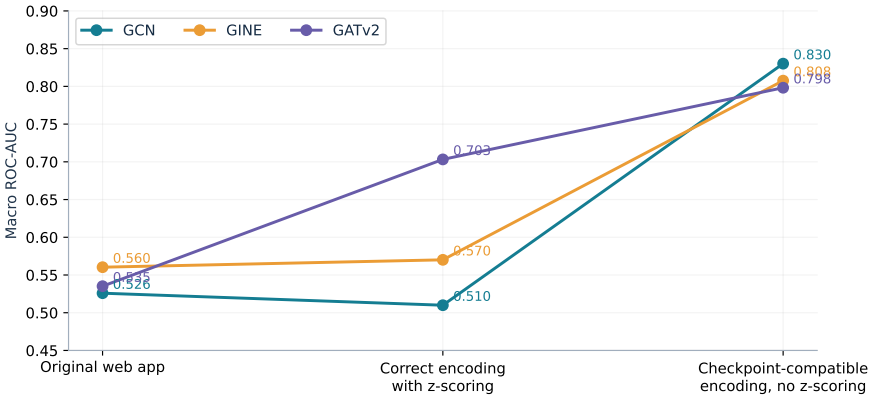


Fig. 1 The same frozen models and recorded test molecules under three preprocessing paths. These are deployment-consistency checks, not separately trained model ablations.

Table 3 NIH external cohort after structure-overlap filtering (n = 598). CI: 95% molecule-bootstrap interval for macro ROC-AUC. Remaining columns are macro-averaged over twelve endpoints; P/R/F1 use threshold 0.5.

Model	ROC-AUC	CI	AP	P	R	F1
GCN	0.795	[0.764, 0.824]	0.259	0.176	0.641	0.270
GINE	0.787	[0.752, 0.817]	0.254	0.164	0.710	0.257
GATv2	0.776	[0.734, 0.812]	0.276	0.162	0.657	0.254
ECFP-RF	0.753	[0.717, 0.788]	0.233	0.308	0.210	0.226
ECFP-LR	0.681	[0.630, 0.734]	0.222	0.214	0.370	0.263

5.2 External Assay Evaluation

Table 3 reports the additional NIH evaluation together with same-training-partition fingerprint baselines. All metrics use observed NIH assay labels rather than presumed toxicity categories attached to familiar chemical names. Figure 2 compares discrimination across cohorts with conditional bootstrap intervals. External scores must be interpreted with their class prevalence, threshold behavior, and structural overlap checks; improved or reduced AUC alone does not explain a biological mechanism.

The highest external macro ROC-AUC is obtained by GCN (0.795); GCN obtains 0.795 and macro AP 0.259. The paired 95% bootstrap intervals for GCN minus GINE and GCN minus GATv2 are [-0.010, 0.026] and [-0.006, 0.045]; both include zero. GATv2 has higher macro AP than GCN (0.276 versus 0.259), so the architecture ranking depends on the metric. This comparison does not justify claiming neural superiority over fingerprint models. Differences in regularization, parameter count, bond-feature access,

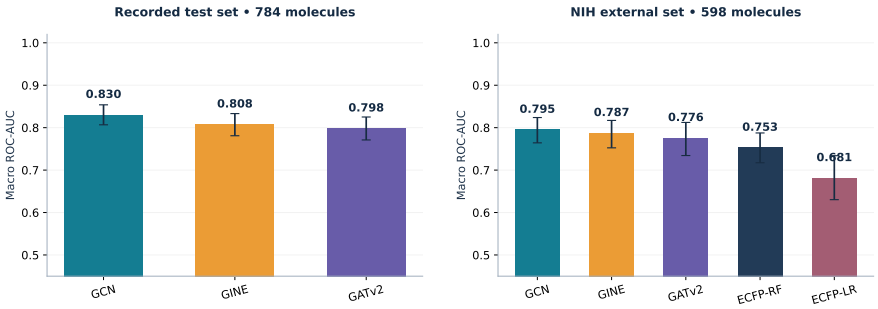


Fig. 2 Internal and external macro ROC-AUC. Error bars are paired molecule-bootstrap percentile intervals. External fingerprint baselines share the recorded training partition; neural models are frozen original checkpoints.

and optimization can contribute to model ranking; no causal architectural ablation isolates these factors.

6 Interface and Reproducibility

The Flask application accepts one SMILES string or a CSV column named `SMILES`, validates structures with RDKit, encodes graphs with PyG, and evaluates one or all three in-memory models. The single-model workflow returns a two-dimensional depiction and an optional generated three-dimensional conformer for visualization. The conformer is not an input to inference. A comparison request displays three separate endpoint-score vectors; it does not average them into an ensemble. CSV output labels are "assay active" or "assay inactive" at the exploratory threshold, replacing unsupported "safe/toxic" labels.

Input tests cover malformed SMILES, empty input, unsupported model names, zero-bond molecules, disconnected salts, mixed-validity CSV files, and batch-size enforcement. Twenty-seven API checks pass. Nine equivalent-SMILES comparisons across three molecules and three models agree within 2×10^{-6}, supporting graph-permutation consistency. For aspirin requests, median end-to-end latencies were GCN: 302.9 ms, GINE: 265.0 ms, GATv2: 226.5 ms. These local measurements include RDKit depictions and HTTP overhead; they do not establish production throughput. No multi-user usability study or production concurrency benchmark is claimed.

The source repository is `https://github.com/eliot-99/Graph-ML-Enabled-Molecular-Design-Assistant-using-Graph-Neural-Networks`. The audit starts from commit `e705e8f7e02a`; local

service corrections, split manifests, per-task predictions, checkpoint SHA-256 hashes, software versions, and the detailed test report accompany the revised source package. These local corrections are not described as already published in the remote repository. Audit inference uses CPU, four PyTorch threads, PyTorch 2.15.0, PyG 2.8.1, and RDKit 2026.09.2.

7 Limitations and Future Work

The retained historical split permits scaffold overlap and its class weights access held-out prevalence. Original neural seed control and independent repeated training are unavailable, so no neural mean±standard-deviation result is claimed. Bootstrap intervals cannot replace five-seed retraining or demonstrate robustness under a new scaffold split. A rigorous next experiment should preserve complete Bemis–Murcko groups within partitions, release group assignments and label counts, learn preprocessing and class weights from training data only, and repeat every GNN and baseline under matched tuning budgets.

The external set is a retrospective public assay dataset, not a prospective laboratory study or clinical validation. Fragment-parent deduplication does not resolve all tautomers, assay artifacts, or chemical-domain shifts. Rare active labels make precision sensitive to prevalence; calibration and task-specific thresholds need validation without consulting external labels. Two-dimensional graphs omit conformational and exposure information. Generated molecular depictions are not explanations, and no validated atom-attribution method is implemented. No claim is made that the assistant reduces laboratory experiments or supports autonomous chemical safety decisions.

8 Conclusion

The study implements and audits three GNN-based Tox21 screening models. GCN's recorded-test macro ROC-AUC of approximately 0.830 is reproduced, while a feature mismatch explains the much lower performance of the original deployed pipeline. Corrected inference, observed-label external testing, controlled fingerprint baselines, and expanded metrics improve the evidence for the screening implementation. The conclusions remain conditional on frozen checkpoints and retrospective assays; scaffold-disjoint, repeated-

seed training and prospective validation are necessary before stronger generalization or practical-impact claims.

References

1. Wu, Z., et al.: MoleculeNet: a benchmark for molecular machine learning. Chemical Science 9, 513–530 (2018). https://doi.org/10.1039/C7SC02664A
2. Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E.: Neural message passing for quantum chemistry. In: ICML, PMLR 70, pp. 1263–1272 (2017).
3. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: ICLR (2017).
4. Xu, K., Hu, W., Leskovec, J., Jegelka, S.: How powerful are graph neural networks? In: ICLR (2019).
5. Hu, W., Liu, B., Gomes, J., Zitnik, M., Liang, P., Pande, V., Leskovec, J.: Strategies for pre-training graph neural networks. In: ICLR (2020). https://arxiv.org/abs/1905.12265
6. Brody, S., Alon, U., Yahav, E.: How attentive are graph attention networks? In: ICLR (2022). https://arxiv.org/abs/2105.14491
7. Yang, K., et al.: Analyzing learned molecular representations for property prediction. Journal of Chemical Information and Modeling 59, 3370–3388 (2019). https://doi.org/10.1021/acs.jcim.9b00237
8. Xiong, Z., et al.: Pushing the boundaries of molecular representation for drug discovery with the graph attention mechanism. Journal of Medicinal Chemistry 63, 8749–8760 (2020). https://doi.org/10.1021/acs.jmedchem.9b00959
9. Rollins, Z.A., Cheng, A.C., Metwally, E.: MolPROP: Molecular Property prediction with multimodal language and graph fusion. Journal of Cheminformatics 16, 56 (2024). https://doi.org/10.1186/s13321-024-00846-9
10. Praski, M., Adamczyk, J., Czech, W.: Benchmarking pretrained molecular embedding models for molecular representation learning. arXiv:2508.06199v5 (2026). https://arxiv.org/abs/2508.06199v5
11. Huang, R., et al.: Tox21Challenge to build predictive models of nuclear receptor and stress response pathways as mediated by exposure to environmental chemicals and drugs. Frontiers in Environmental Science 3, 85 (2016). https://doi.org/10.3389/fenvs.2015.00085
12. Bemis, G.W., Murcko, M.A.: The properties of known drugs. 1. Molecular frameworks. Journal of Medicinal Chemistry 39, 2887–2893 (1996). https://doi.org/10.1021/jm9602928
13. PyTorch Geometric contributors: SMILES utilities and convolution documentation. https://pytorch-geometric.readthedocs.io/en/latest/_modules/torch_geometric/utils/smiles.html. Accessed 21 Nov 2026.
14. RDKit contributors: MurckoScaffold API reference. https://www.rdkit.org/docs/source/rdkit.Chem.Scaffolds.MurckoScaffold.html. Accessed 21 Nov 2026.
15. NIH NCATS: Tox21 Data Challenge 2014, data and resources; final-evaluation structures and observed assay labels. https://tripod.nih.gov/tox21/challenge/data.jsp. Accessed 21 Nov 2026.