Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Negative examples in protein–protein and drug–target interaction prediction: a critical review of training strategies and evaluation practices

Justin Li

jli89@tulane.edu

Tulane University

Systematic Review

Keywords: negative sampling, positive-unlabelled learning, virtual screening, data leakage, protein–protein interaction, drug–target interaction

Posted Date: November 9th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11331994/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Negative examples in protein–protein and drug–target interaction prediction: a critical review of training strategies and evaluation practices

Justin Li

Center for Biomedical Informatics and Genomics, Tulane University, New Orleans, LA, USA

Corresponding author: Justin Li, jli89@tulane.edu

Acknowledgements

The author thanks Ray Su for his guidance and suggestions, and Hong-Wen Deng for his support.

Abstract

Models for protein–protein and drug–target interaction prediction usually rely on constructed negative examples because experimentally supported non-interactions are scarce. Unobserved pairs may be sampled at random, filtered by biological rules or replaced with presumed-inactive decoys. These choices affect the patterns a model learns and the difficulty of the benchmark used to assess it. This critical review brings together evidence from interaction prediction, virtual screening and positive-unlabelled learning to examine how negative construction influences model evaluation. We distinguish the evidence supporting a negative label from the rule used to select a pair, and compare the rationale, empirical support and limitations of common selection strategies. Across these studies, more confident negative labels do not necessarily produce more informative tests: selection can make negatives easier to distinguish, amplify degree-related shortcuts or introduce decoy bias. The distinction between training and evaluation helps explain some apparently conflicting findings. A sampling rule that assists learning may give a misleading performance estimate when it also defines the test set. We use this distinction to develop conditional recommendations for selecting negatives, choosing data splits and reporting class ratios and baseline results. A reporting checklist also identifies information that experimental groups can retain to make inactive measurements reusable. We also relate these findings to negative sampling in link prediction, recommendation and knowledge-graph completion. Explicit descriptions of both training and test negatives are needed to interpret reported gains and assess their relevance to the intended biological application.

Keywords: negative sampling; positive-unlabelled learning; virtual screening; data leakage; protein–protein interaction; drug–target interaction

1 Introduction

1.1 Interaction data and missing negatives

Machine-learning models are widely used to propose new protein–protein interactions (PPIs) and drug–target interactions (DTIs) for experimental follow-up. Positive examples come from curated databases and large screens; a single systematic two-hybrid map of the human proteome, for instance, reported about 53,000 binary interactions [1]. Confirmed non-interactions are far scarcer. Curated collections of non-interacting protein pairs contain a few thousand entries [2, 3]. Bioactivity databases such as ChEMBL and BindingDB record quantitative measurements [4, 5], from which inactive pairs can be defined only by choosing an activity threshold [6].

Interaction maps are also far from complete. Estimates of the size of the human interactome have ranged from about 130,000 binary interactions [7] to 154,000–369,000 interactions, and in 2006 the human map was judged to be roughly 10% complete [8]. An unobserved pair is therefore not a confirmed negative.

Most studies therefore construct negatives by sampling unobserved pairs, applying biological filters or selecting presumed-inactive decoys. Although often described only briefly, the construction procedure affects both the training data and the comparison on which a model's reported performance rests.

1.2 Recent evidence on negative construction

Studies since 2022 have examined several consequences of negative construction. In PPI prediction, selecting test negatives by sequence dissimilarity made the task easier and inflated accuracy estimates [9]. In compound–protein prediction, choices about negative synthesis and evaluation also produced over-optimistic estimates; under stringent evaluation protocols, a simple kernel baseline outperformed state-of-the-art methods [10].

Related work has examined what interaction predictors learn from these datasets. Deep models of protein–ligand binding exploited node degree in the interaction network [11], while deep sequence-based PPI predictors fell to random performance when sequence similarity between training and test proteins was controlled [12]. Analyses across several biomolecular interaction

tasks also found that the scale-free topology of interaction networks can distort performance assessment [13].

The consequences extend to comparisons between models. In a recent PPI benchmark, replacing random negatives with degree-matched negatives changed which models ranked best [14]. Together, these studies motivate closer attention to the construction of the negative set when interpreting a reported gain.

1.3 Previous reviews

Previous reviews provide much of the background for this discussion (Table 1). Jansen and Gerstein [15] argued that gold-standard negatives matter as much as positives and that evaluation must account for the expected ratio of positives to negatives. Subsequent reviews have generally approached negative examples within a particular prediction task or methodological literature.

Reviews of PPI prediction cover algorithms, validation schemes, metrics and databases [16, 17]. DTI reviews discuss data, methods and empirical comparisons [18–21], as well as the construction of benchmark datasets [22]. Within this literature, Chen et al. [18] contrasted supervised and semi-supervised models according to their use of negative samples, and Cheng et al. [19] identified the incorporation of negative samples as an open challenge.

Virtual-screening reviews examine decoy selection, from randomly chosen compounds to experimentally validated inactives [23, 24], and consider decoys in the wider context of docking validation [25]. Reviews of positive-unlabelled (PU) learning address assumptions, algorithms and evaluation, including applications to interaction prediction [26, 27]. A further body of work examines data leakage and evaluation design in biological machine learning [28–31].

Several recent papers focus more directly on negatives: an opinion paper on PPI and host–pathogen prediction [9], an analysis of compound–protein evaluation [10], and a PPI benchmarking framework that distinguishes balanced sampling for training from sampling for evaluation [32]. Each concentrates on one interaction type, leaving useful connections to be made across these settings.

Outside biology, negative sampling has been surveyed for recommendation [33] and for knowledge-graph representation learning [34]. Both surveys group sampling methods by technique, largely as part of model training. The present review differs in two ways. It organises

negatives by the evidence behind their labels and by whether they serve training or evaluation. It also draws on PPI and DTI studies, where some negatives have been measured experimentally.

1.4 Perspective of this review

This review examines negative construction across PPI prediction, DTI prediction and virtual screening, with particular attention to the purpose of each negative set. Park and Marcotte [35] distinguished training negatives from evaluation negatives in PPI prediction, and the distinction has been restated more recently [32]. Applying it across these literatures helps explain why studies can reach different conclusions about the same sampling strategy. The disagreement over degree-balanced sampling, discussed in Section 3.4, illustrates this point.

We describe negative sets in terms of their label evidence and selection rule (Section 2, Fig. 1), then compare the empirical support and limitations of common strategies (Section 3, Table 2). The discussion of training and evaluation in Section 4 leads to recommendations and a reporting checklist in Section 5, including guidance for experimental groups seeking to make their negative results reusable. Section 6 compares these findings with recommendation, knowledge-graph completion and graph link prediction (Table 4).

1.5 Scope and literature selection

We cover binary prediction of PPIs and DTIs, including compound–protein interactions (CPIs). Affinity regression is discussed only where a threshold turns measurements into negatives. Virtual screening for a given target is treated separately, because it asks a narrower question than pairwise DTI prediction. Other interaction types, such as RNA–protein or microRNA–disease associations, are outside our scope.

This is a critical narrative review, not a systematic review. On 4 October 2026 we searched PubMed titles and abstracts for terms describing negatives (negative sampling, negative examples or samples, non-interacting, negative data, decoys, positive-unlabelled) combined with interaction tasks (protein–protein, drug–target, compound–protein and protein–ligand interaction; virtual screening). The search returned 874 records, 35 of them indexed as reviews. We screened the 35 reviews and used targeted searches for specific strategies, datasets and benchmarks to select studies that proposed, compared or criticised a way of constructing negatives, or measured its consequences. We also included methodological work from machine learning, on positive-

unlabelled learning, evaluation metrics and leakage, where it bears directly on these questions. For Section 6, studies from link prediction, recommendation and knowledge-graph completion were selected because they address the same questions; that literature was not searched systematically. Where several studies made the same point, we cite the one that introduced it or gave the most direct evidence. The selection is illustrative rather than exhaustive.

Table 1. Previous reviews and closely related studies.

Work	Type	Domain	Main focus	Relation to this review
Jansen and Gerstein [15]	Review	PPI	Gold-standard positives and negatives; prior ratio; localisation-based negatives	Early statement of the problem; we extend it to DTI and to evaluation design
Hu et al. [16]; Soleymani et al. [17]	Reviews	PPI	Algorithms, validation schemes, metrics, databases; deep learning	We focus on how negatives are built and used
Chen et al. [18]; Cheng et al. [19]; Ezzat et al. [20]; Bagherian et al. [21]	Reviews, empirical survey	DTI	Data, methods, empirical comparison; supervised vs semi-supervised use of negatives; negatives as an open challenge	We compare negative-construction choices across PPI and DTI by purpose
Atas Guvenilir and Doğan [22]	Benchmark study	DTI	Featurisation, data preparation, benchmark datasets	We focus on the negatives within such datasets
Lagarde et al. [23]; Réau et al. [24]; Kittelson et al. [25]	Reviews	Virtual screening, docking	Selection of actives and decoys; decoys in docking validation	We relate decoys to pairwise DTI and PPI negatives and separate the two tasks
Bekker and Davis [26]; Li et al. [27]	Reviews	PU learning	Assumptions, methods, evaluation; bioinformatics applications	We place PU learning beside negative-selection strategies and their evaluation
Walsh et al. [28]; Whalen et al. [29]; Kapoor and Narayanan [30]; Bernett et al. [31]	Guidelines, reviews	Machine learning in biology	Validation, pitfalls, data leakage	We separate construction bias from leakage
Ma et al. [33]; Madushanka and Ichise [34]	Reviews	Recommendation, knowledge graphs	Negative-sampling methods grouped by technique, largely for training	Same problem outside biology; we organise negatives by label evidence and purpose (Section 6)
Lannelongue and Inouye [32]	Benchmarking framework	PPI	Pitfalls, network topology and hubs; balanced sampling for training but not for evaluation	Same distinction for PPI; we extend it to DTI and virtual screening and attach conditions
Neumann et al. [9]	Opinion with analysis	PPI, host–pathogen	Similarity-filtered negatives; training-only versus test use	Closest in spirit; we add DTI and virtual screening, two axes and guidance
Yaseen et al. [10]	Analysis	Compound–protein	Negative synthesis, similarity control, evaluation aligned with use	DTI counterpart; we connect it with PPI and screening evidence
Najm et al. [36]; Chatterjee et al. [11]; Li et al. [13]; Cabrera-Andrade et al. [14]	Empirical studies	DTI, PPI, several	Balanced sampling, network shortcuts, degree-balanced sampling, multiple negative rules	We compare their mechanisms within one framework

2 Evidence and selection in negative construction

Two attributes are needed to describe a negative set: the evidence supporting its labels and the procedure used to select its members. These attributes need to be considered separately. Figure 1 uses them as two axes for organising the resources and studies reviewed here.

2.1 Label evidence

Measured. An experiment tested the pair and did not detect an interaction or activity:

- inhibitor–kinase combinations measured in profiling panels [37];
- compounds recorded as inactive in screening campaigns [38, 39].

Negatives derived from large two-hybrid screens are discussed in Section 3.6 rather than listed here, because whether a given pair was actually tested must itself be inferred [40].

The interpretation of a measured negative depends on the assay. Non-detection is limited by assay and sampling sensitivity [7], while an inactive label depends on the activity threshold and experimental conditions [6, 41].

Inferred. The label comes from reasoning about a specific pair rather than from testing it. An example is two proteins that belong to a solved complex but make no contact in its three-dimensional structure [2].

Curation describes how a resource is assembled, but does not establish the evidence for every label. The Negatome database, for example, contains both literature reports with experimental support and structure-based inferences [2, 3]. The two subsets therefore require different descriptions of their label evidence.

Unlabelled. The pair simply has no recorded interaction. It may never have been tested, or it may have been tested without the result being reported. How many hidden positives such pairs contain depends on how complete the interaction map is [8]. It also depends on how intensively the entities involved have been studied, because the number of recorded partners of a protein reflects how often it has been screened [42].

2.2 Selection rule

The second axis describes how pairs enter the negative set:

- **All tested pairs**, as in a complete profiling panel.
- **Uniform random sampling** of unobserved pairs.
- **Filters** that keep only pairs judged unlikely to interact, for example proteins in different subcellular compartments [15] or pairs dissimilar to known partners [43, 44].
- **Balancing** so that each entity appears as often among negatives as among positives [36, 45].
- **Property matching**, which selects decoys that resemble active compounds in physicochemical properties [46, 47].

Filters apply assumptions about which pairs are unlikely to interact. A localisation filter, for example, changes which unlabelled pairs enter a dataset without providing a measurement for any individual pair. Their label evidence remains unlabelled.

2.3 Combining the axes

Most resources occupy one cell of Fig. 1, and some span several:

- The LIT-PCBA benchmark, built from PubChem bioassays, combines measured inactives with property-range matching [39].
- The DUD-E decoy set (Directory of Useful Decoys, Enhanced) combines presumed inactives with property matching [47].
- Liu et al. [43] applied a dissimilarity filter to unlabelled pairs.
- The AI-Bind pipeline combined network-derived negatives with experimentally confirmed non-binders in order to balance each node [11].

Figure 1 is a reading aid for the selected literature rather than a new classification system. An empty cell indicates that no included study falls in that combination; it does not establish a research gap.

PU learning does not appear in the grid, because it is not a source of negatives but a way of learning without them (Section 3.7).

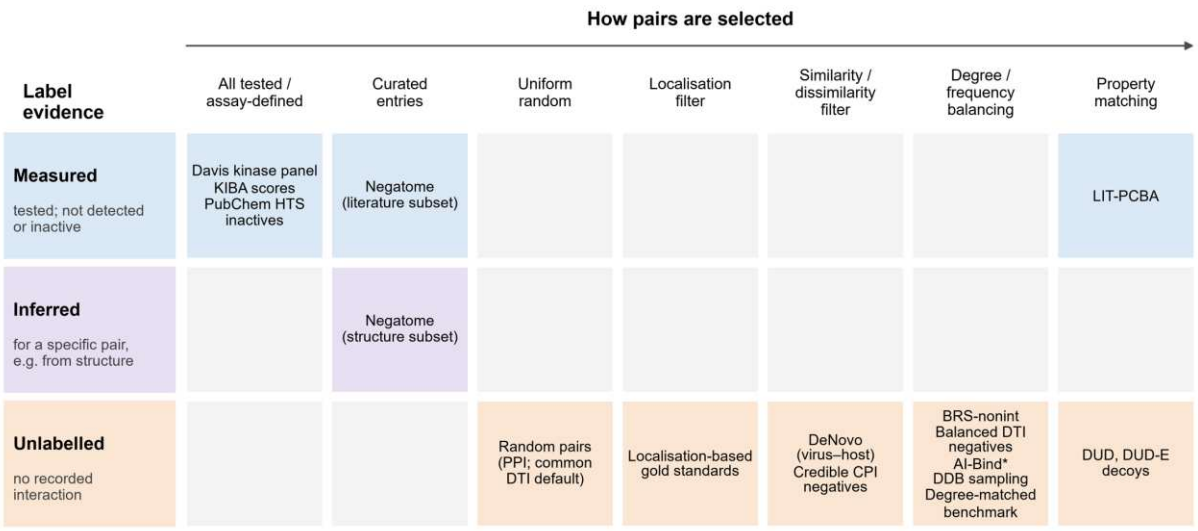


Fig. 1. Two axes for describing negatives in PPI and DTI studies. Rows give the label evidence: measured (tested, not detected or inactive); inferred for a specific pair, for example from structure; unlabelled. Columns give the selection rule: all tested or assay-defined pairs; curated entries; uniform random sampling; localisation filter; similarity or dissimilarity filter; degree or frequency balancing; property matching. Representative resources and studies are placed in the cells:

- Davis kinase panel and KIBA integrated activity scores [37, 41]; PubChem high-throughput screening (HTS) inactives [38];
- Negatome literature and structure subsets [2, 3];
- LIT-PCBA [39];
- DUD and DUD-E decoys [46, 47];
- localisation-based negatives [15, 48];
- dissimilarity-filtered negatives [43, 44];
- balanced negatives [11, 13, 14, 36, 45], where AI-Bind (*) also adds experimentally confirmed non-binders [11];
- random negatives [35, 48], a common default in DTI prediction [49].

The grid is a reading aid for the studies reviewed here. An empty cell means no example among the selected studies and does not by itself indicate a research gap.

3 Negative-selection strategies and their trade-offs

Table 2 compares six negative-selection strategies and positive-unlabelled learning, which offers a related approach to using unlabelled data. The evidence below concerns both confidence in individual labels and changes to the distribution of selected pairs. Improving the former can substantially alter the latter, with different consequences for training and evaluation.

3.1 Random unobserved pairs

The simplest choice samples pairs with no recorded interaction, often among proteins or drugs that already appear in the positive set [48]. In DTI prediction this remains a common default [49]. Its justification is sparsity: in a sparse interaction space, uniformly sampled unobserved pairs may contain relatively few hidden positives [9]. The contamination rate nevertheless depends on the candidate entities and on any sampling restrictions.

A concern with random pairing is the uneven representation of entities in interaction networks. A few proteins and drugs have many recorded partners, whereas most have few. Positive pairs over-represent well-connected entities, but uniformly sampled negatives do not. A classifier can exploit this difference by learning how often each entity occurs among the positives. Hub proteins inflated the reported accuracy of sequence-based PPI predictors [45]. When sequence similarity between training and test proteins was minimised, deep sequence-based PPI models fell to random performance, implicating similarity and node degree in their earlier success [12]. Degree-related shortcuts have also been described for protein–ligand binding [11] and other biomolecular interactions [13]. Several published PPI predictors failed to outperform models using features with no logical link to interaction; the authors attributed this result partly to the uneven study of proteins in the literature [50].

The relevance of degree information depends on the prediction task. Park and Marcotte [35] argued that sequence features and protein "hubbiness" can both contribute to useful prediction, so degree information is not inherently an artefact. In compound–protein prediction, random pairing for training and evaluation produced better generalisation than more elaborate negative-selection schemes [10]. These findings support random negatives as a practical default, accompanied by a degree-only baseline to show how much performance degree alone explains.

3.2 Localisation filters

Early PPI studies selected "gold-standard" negatives from proteins annotated to different subcellular compartments, reasoning that proteins unlikely to meet were also unlikely to interact [15]. Ben-Hur and Noble [51] showed that this restriction made negatives easier to distinguish and biased accuracy estimates upwards for both sequence-based and other predictors. Later users of the strategy acknowledged the problem [48], and the Negatome curators noted that localisation-based negatives tend to contain functionally dissimilar proteins [2]. Greater

confidence in these labels therefore comes with a change in the pairs on which the predictor is assessed.

3.3 Similarity and dissimilarity filters

A related idea keeps only unobserved pairs whose members are dissimilar to known partners. The DeNovo predictor applied it to virus–host PPIs, selecting negatives by sequence dissimilarity to reduce label noise [44]. For compound–protein pairs, Liu et al. [43] used the converse of the similarity principle: a protein dissimilar to every known target of a compound is unlikely to be targeted by it. They reported much higher classifier performance on the screened negatives than on random ones.

Neumann et al. [9] showed that constructing test negatives in this way makes the classification problem easier and over-estimates accuracy. They distinguished this from using filtered negatives only during training, which does not itself leak information. Even in that setting, however, differences between the training and test distributions may outweigh the reduction in label noise. The relevant evidence for a benefit is performance on an independent test set representing the intended application.

3.4 Degree- and frequency-balanced sampling

To remove the degree shortcut, several studies sample negatives so that each entity appears as often among negatives as among positives, or so that the degree distributions match:

- Yu et al. [45] proposed such a balanced negative set for PPIs.
- Najm et al. [36] applied the principle to drug–target pairs. When identifying targets for a given drug, they found fewer false positives among the top-ranked targets.
- AI-Bind combined network-derived negatives with experimentally confirmed non-binding pairs, so that every protein and ligand had both positive and negative examples [11].
- Degree-distribution-balanced sampling was proposed for several interaction types [13].

A recent PPI benchmark showed how much the choice matters. Replacing random with degree-matched negatives lowered the area under the precision–recall curve (AUPRC) of its gradient-boosting model by 0.049–0.259 across datasets and changed the ranking of models [14].

Balanced sampling answers a different question from random sampling, and the two have been confused. Park and Marcotte [35] argued that Yu et al. [45] reached an erroneous conclusion, namely that sequence features are barely better than random, because their balanced test sets did not represent the population of candidate pairs. Lannelongue and Inouye [32] restated the point: balanced sampling helps training but should not be used to evaluate models. We read this as applying to the main performance estimate. Reported alongside random negatives, balanced evaluation remains informative as a diagnostic. A drop in performance when degree is equalised, as reported in a recent PPI benchmark [14], suggests that the model may have used degree-related signal. Because equalising degree also changes which pairs are tested and how hard the task is, such a drop is best read together with a degree-only baseline. Balanced sampling is also appropriate for training when the task is to rank partners of a given entity, as in target identification for a drug [36]. Li et al. [13] trained on balanced negatives but sampled test negatives at random, which keeps the two purposes apart.

3.5 Property-matched decoys in virtual screening

Virtual screening evaluates compounds against a specified target, whereas pairwise DTI prediction spans both compounds and targets. Decoy benchmarks pair active compounds with presumed inactives that have similar physicochemical properties but different molecular topologies. Property matching is intended to limit enrichment driven by gross property differences, while topological differences reduce the likelihood that the decoys bind [46, 47]. These are presumed inactive compounds, without measured evidence of inactivity.

Reusing decoy benchmarks for machine learning exposed biases in their construction. Convolutional models trained on DUD-E exploited analogue and decoy bias rather than protein–ligand interactions [52], and models also learned biases in other standard structure-based benchmarks [53]. In ligand-based prediction, redundancy between training and validation compounds, including both actives and inactives, explained much of the reported performance [54].

LIT-PCBA addressed these problems using confirmed actives and inactives from dose–response PubChem assays, restricting both groups to similar property ranges and pruning residual bias [39]. Decoy sets remain useful for evaluating docking against a particular target. Their

interpretation should remain specific to that task when considering negatives for general pairwise DTI prediction.

3.6 Experimentally supported negatives

PPIs.

- Negatome collects non-interacting pairs from two sources with different kinds of evidence [2, 3]:
  - pairs reported as non-interacting in the literature with experimental support;
  - pairs inferred from three-dimensional structures of complexes in which the two proteins do not touch.

  Its curators note a clear bias toward well-studied cases [3].
- Large two-hybrid screens offer another source: pairs that were probably tested but not observed [40]. Whether a pair was tested is itself an inference. Non-detection also reflects the sensitivity of the assay and of sampling, so it is evidence of absence rather than proof [7].

DTIs.

- A kinase profiling panel measured every combination of 72 inhibitors and 442 kinases [37].
- Public high-throughput screens record inactive compounds alongside actives validated in confirmation screens [38]. LIT-PCBA retains confirmed inactives from dose–response assays [39].
- Integrated activity scores such as KIBA combine heterogeneous IC50, Ki and Kd measurements. Only about 1% of compound–kinase pairs in the released matrix have a score, so low activity, missing values and untested pairs must be kept apart [41].

These resources require assay definitions and thresholds to accompany their labels. Prospective testing provides additional evidence about predictive performance: Cichonska et al. [55] predicted unmeasured kinase inhibitor affinities, assayed 100 compound–kinase pairs and reported a correlation of 0.77 between predicted and measured values.

3.7 Positive-unlabelled learning

Positive-unlabelled (PU) learning uses the distinction between observed positives and unlabelled pairs without assigning a negative label to every unlabelled example. Two broad approaches are relevant here:

- **Two-step methods** first extract "reliable" negatives from the unlabelled set and then train a standard classifier. The PUDTI method took this route for DTIs [49].
- **Direct methods** learn from positives and unlabelled data under an assumption about how the positives came to be labelled:
  - the classic assumption is that labelled positives are a random sample of all positives, "selected completely at random" (SCAR) [56];
  - bagging over random subsamples of the unlabelled set is another widely used approach [57];
  - the wider literature covers methods for other labelling assumptions [26, 27].

The SCAR assumption warrants scrutiny in interaction data because the number of recorded partners of a protein depends on how often it has been screened [42]. This association does not establish a violation in any particular dataset, but it does mean that random labelling should be justified when assumed.

PU learning also provides methods for evaluation when some unlabelled pairs are positive. Treating all such pairs as negatives biases performance estimates, such as the area under the receiver operating characteristic (ROC) curve. Corrections are possible under specified labelling assumptions if the class prior is known or can be estimated; where labelled positives also contain errors, the noise rate is required as well [58, 59].

333 **Table 2. Strategies for selecting negative pairs in PPI and DTI prediction.** Label evidence and selection correspond to the two axes of Fig. 1.

Strategy	Label evidence	How pairs are selected	Representative uses	Designed to address	Documented limitations	Appropriate use
Random unobserved pairs	Unlabelled	Uniform sampling of unobserved pairs, often among entities present in the positive set [48]	PPI [35, 48]; DTI/CPI, a common default [10, 49]	Scale; few hidden positives in a sparse interaction space [9]	Positives and negatives differ in node degree, so models can score well from degree alone [11–13, 45]; some published methods did not beat "illogical"-feature controls [50]	Default when the intended use is screening of all candidate pairs, where degree can be genuinely informative [35]; in CPI, random pairing generalised better than more elaborate schemes [10]; report degree-only baselines
Localisation filter	Unlabelled; filter assumes differently localised proteins rarely interact	Pairs of proteins annotated to different compartments [15]	PPI [15, 48]	Higher-confidence negatives	Constrains the negative distribution, makes the task easier and inflates accuracy estimates [51], as later acknowledged [48]; skews negatives toward functionally dissimilar pairs [2]	May be considered for training if the benefit is shown on an independent test set that matches the intended use; a test set built this way supports conclusions only about the restricted pairs
Similarity or dissimilarity filter	Unlabelled	Keep unobserved pairs dissimilar to known partners or targets (sequence or multi-source similarity)	Virus–host PPI [44]; CPI [43]	Fewer hidden positives	Test sets built this way are easier than the real task and over-estimate accuracy [9]; higher scores on screened than on random negatives [43] are therefore not, on their own, evidence of better generalisation	May be used for training; confirm any gain on an independent test set that matches the intended use, because the train–test distribution shift can lower accuracy [9]
Degree- or frequency-balanced sampling	Unlabelled; sometimes combined with measured non-binders [11]	Each entity appears as often in negatives as in positives, or the degree distribution is matched	PPI [14, 45]; DTI [11, 36]; several interaction types [13]	Remove the degree shortcut	Answers a different question from random sampling: judging population-level performance on balanced test sets produced a disputed conclusion [35, 45]; balanced sampling helps training but should not supply the main performance estimate [32]; model rankings change between random and degree-matched negatives [14]	Diagnostic or control evaluation reported alongside random negatives; training option for ranking partners of a given entity [36]; balanced training can be paired with random test negatives [13]
Property-matched decoys (target-specific virtual screening)	Unlabelled: decoys presumed inactive	Match the physicochemical properties of actives while maximising topological dissimilarity [46, 47]	Docking and virtual-screening benchmarks (DUD, DUD-E); not used for PPI	Prevent enrichment driven by gross property differences	Analogue and decoy biases, and machine-learning models can learn the construction rule [52, 53]; training–validation redundancy inflates ligand-based benchmarks [54]; DUD, DUD-E and MUV judged biased [39]	Benchmarking screening for a given target, with bias checks; not as negatives for general pairwise DTI models; for machine learning, prefer confirmed inactives [39]
Experimentally supported negatives	Measured (tested, not detected or inactive) or inferred from structure; classified per subset	All measured pairs in a panel, assay-defined inactives, or curated entries	PPI: Negatome literature and structure subsets [2, 3]; two-hybrid pairs probably tested but not observed, with testing itself inferred [40]. DTI: kinase panel [37]; integrated scores [41]; PubChem HTS inactives [38]; confirmed inactives [39]	More direct label evidence	Small and biased toward well-studied cases [3]; few target classes; labels depend on thresholds and assay conditions; non-detection reflects assay and sampling sensitivity [7]; integrated scores leave most pairs unmeasured [41]	Prefer when the assay and threshold match the intended use; report assay type, thresholds and the tested space

Strategy	Label evidence	How pairs are selected	Representative uses	Designed to address	Documented limitations	Appropriate use
Related approach: PU learning (not a source of negatives)	Unlabelled pairs treated as a mixture of positives and negatives	Two-step extraction of reliable negatives [49], or learning positives versus unlabelled under a labelling assumption [26, 56, 57]	DTI [49]	Avoid treating unlabelled pairs as negatives	Results depend on the labelling assumption, e.g. SCAR [56], which study bias makes questionable [42]; naive ROC and precision–recall estimates from positive-versus-unlabelled data are biased [58, 59]	When the labelling mechanism or class prior can be justified or estimated; under stated labelling assumptions, class priors (and label-noise rates) allow corrected performance estimates [58, 59]

4 Negative examples in training and evaluation

4.1 Two purposes

Park and Marcotte [35] distinguished two reasons for sampling negatives:

- **Training negatives** should help a model learn a useful decision rule, and they may be biased if that helps.
- **Test negatives** should allow the measured performance to be generalised to the pairs the model will be applied to, so they should be unbiased with respect to those pairs.

This distinction helps interpret the strategies in Section 3. A selection rule may be useful during training yet yield a misleading performance estimate if it also determines which negatives appear in the test set.

Hamp and Rost [60] further showed that the comparison depends on the relationship between the training and test data:

- performance depends on how similar test proteins are to the training negatives, not only to the training positives;
- which method appears best depends on that similarity, on how many true interactions should be found, and on the expected ratio of negatives to positives.

4.2 Sharing a construction rule between training and test

A shared construction rule becomes problematic when it produces differences between positives and negatives that a model can exploit but will not encounter in use. Localisation filters [51], similarity filters [9] and decoy generation [52, 53] can create such differences. This is a form of shortcut learning, in which a decision rule performs well on the benchmark but fails to transfer to more realistic conditions [61]. Using the same rule for training and testing is therefore not, by itself, sufficient to diagnose the problem.

We call this construction bias rather than leakage. Leakage refers to information flowing illicitly between training and test data, which yields performance estimates better than those seen in the intended application [30, 31]. Construction bias can arise even when training and test sets are perfectly separated.

4.3 Splitting first, then constructing negatives

For pairwise prediction, a split defines which combinations of familiar and unseen entities are evaluated.

- **PPI.** Park and Marcotte [62] classified test pairs by whether both proteins (C1), one protein (C2) or neither (C3) appear in the training set. More than 99% of the test pairs in typical cross-validation fell into C1, yet C1 pairs are a minority of all possible human protein pairs. Cross-validated performance therefore did not describe the other two classes.
- **DTI.** Pahikkala et al. [6] made the same distinction, according to whether training and test sets share drugs, targets, both or neither, and listed it among four factors that can produce over-optimistic results.

Entity overlap is not an error in itself. If the intended use is to find new interactions among already-studied proteins, C1-type evaluation is appropriate, provided this limitation is stated [62]. The error is a mismatch between the split and the intended use. Each of the following answers a specific question:

- strict splits that exclude proteins similar to those seen in training [12, 63];
- cluster-based splits that keep compound series together [64];
- time splits that mimic prospective prediction [65].

Negatives should be built after the split, within the candidate pairs that each partition allows. Each construction step should state which data it uses and whether those data would be available in the intended application:

- **Network-based or fitted rules** (degree balancing, network-derived negatives, two-hop "hard" negatives) must not use evaluation labels; in practice their statistics are usually taken from the training side. One recent benchmark, for example, defined its two-hop negatives using the training-positive graph only [14].
- **Test-set construction** necessarily uses evaluation data, but none of that information should feed back into model training, preprocessing fits or model selection.

The information available to a model also differs between transductive and inductive settings. In a transductive setting, all entities are present during training, including those later evaluated. In an inductive setting, test entities are absent from training, although their features are available

when predictions are made [66]. Access to a test entity is therefore not automatically leakage; the relevant question is whether that information would be available in the stated application.

4.4 Class ratio and metrics

The ratio of negatives to positives in a test set is also a design choice.

- **AUROC (area under the ROC curve)** does not depend on this ratio as long as the score distributions within each class stay the same. It does depend on which negatives are chosen, so it is not invariant to the construction rule.
- **Precision-based measures** depend directly on the ratio. Precision–recall curves show more directly the trade-off between retrieving positives and keeping predictions precise, which matters when negatives greatly outnumber positives [67]. The two kinds of curve are nonetheless linked: a curve dominates in ROC space if and only if it dominates in precision–recall space [68].

The true ratio is rarely known:

- The PPI predictor D-SCRIPT chose ten negatives per positive on the grounds that true interactions are rare [69].
- Published PPI performance was overstated when evaluation used a higher proportion of positives than occurs naturally [50].

When the realistic ratio is unknown, results should be reported across several plausible ratios.

Two further cautions apply:

- **Sampled ranking metrics.** In recommender systems, metrics computed on a sample of candidates were shown to be inconsistent with their full versions, to the point that a comparison between two methods need not hold even in expectation [70].
- **Incomplete positives.** When some positives are unlabelled, PU corrections can recover performance estimates under the stated labelling assumptions, given the class prior [58, 59].

Figure 2 places these choices along a typical PPI or DTI pipeline.

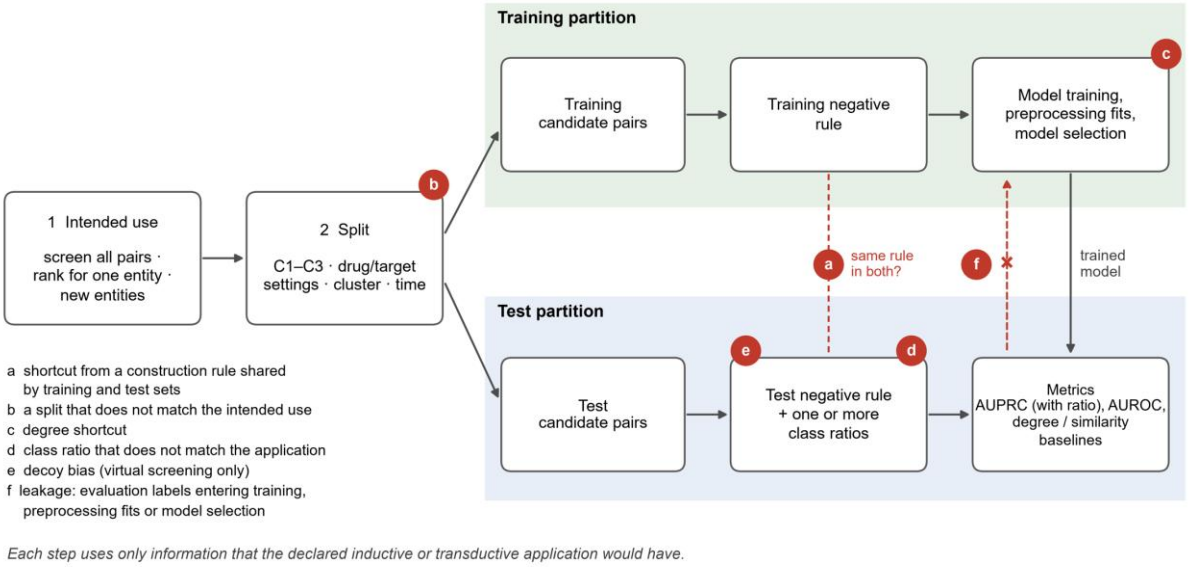


Fig. 2. Where negative choices enter a PPI or DTI pipeline. The intended use determines the split (C1–C3, drug/target settings, cluster or time). Negatives are then built separately within the training and test partitions, and test negatives are drawn with one or more class ratios before metrics are computed. Each step uses only information that the declared inductive or transductive application would have; evaluation labels never enter training, preprocessing fits or model selection. Markers show where documented problems enter:

- (a) a shortcut created by a construction rule shared by training and test sets;
- (b) a split that does not match the intended use;
- (c) degree shortcuts;
- (d) a class ratio that does not match the application;
- (e) decoy bias (virtual screening only);
- (f) leakage, when test information enters training, preprocessing fits or model selection.

5 Choosing, evaluating and reporting negatives

5.1 Start from the intended use

The biological question determines which candidate pairs belong in the test set. Three common settings illustrate the implications for splitting and negative selection:

1. **Screening all candidate pairs**, for example to propose new interactions across a proteome or a compound library. Test negatives should resemble the candidate pairs: random unobserved pairs, with results reported separately for C1, C2 and C3 pairs, or for the DTI equivalents [6, 62].

2. **Ranking candidates for a given entity**, such as targets for a drug or active compounds for a target. Training schemes, including balanced sampling, can be compared for this task [36]. Evaluation, however, should keep an explicit set of candidate targets or compounds, state the label evidence for each, and use ranking metrics over that set; Najm et al. [36] assessed where the true targets ranked among the candidates. Balanced test negatives do not by themselves represent the candidate set. Decoy benchmarks apply only to virtual screening and require bias checks (Section 3.5).
3. **Predicting for new proteins, drugs or species**, for example across species [69] or for novel viruses [44]. This requires splits that keep the new entities out of training. When the goal is to predict for proteins of low sequence similarity or compounds with new scaffolds, similar entities must be excluded as well [12, 63, 64].

5.2 Choosing training negatives

Training negatives can be selected to improve learning, even if their distribution differs from that expected in use (Section 4.1). The following choices require evidence appropriate to the task:

- **Measured inactives**, when they exist and their threshold matches the intended use. Examples are kinase panels [37] and confirmed screening inactives [39].
- **Random unobserved pairs** otherwise, reported with a degree-only baseline [10, 35].
- **Localisation or similarity filters** may be considered for training, provided their benefit is demonstrated on an independent test set appropriate to the intended application [9]. A test set built with the same filter supports conclusions only about the restricted pairs [51].
- **Balanced sampling** for per-entity ranking tasks, or as a robustness check [13, 36].
- **PU methods** when the labelling mechanism or the class prior can be argued [26].

5.3 Evaluating

The following checks help distinguish gains in prediction from advantages introduced by the evaluation design:

1. Report results under more than one negative rule where the rule could matter. When degree differences could create a shortcut, add degree-matched negatives as a control, and add

measured negatives where they exist. A recent benchmark recommends interpreting models across such rules rather than under a single one [14].

2. Include baselines that use only node degree or only sequence or chemical similarity, together with a control model built on uninformative features [12, 50]. If a proposed model does not beat these baselines, its advantage is unproven.
3. Choose the split from the intended use (Section 4.3), and report the C1/C2/C3 or drug/target breakdown.
4. Report performance at several plausible class ratios when the true ratio is unknown (Section 4.4).
5. Repeat the negative sampling and report the variability across draws. Najm et al. [36], for example, generated five negative sets per scheme.

5.4 Reporting

Table 3 lists the information needed to interpret a negative set and its use in evaluation. It complements the DOME recommendations for data, optimisation, models and evaluation in biological machine learning [28], and the guiding questions on leakage [31], with details specific to negative construction.

Table 3. Reporting checklist for negatives in PPI and DTI studies.

#	What to report	Why it matters	Evidence
1	Source, version and filters of the positive pairs	Positives define which entities, and how many partners per entity, the negatives are contrasted with	[12, 42, 45]
2	Label evidence of the negatives (measured, inferred or unlabelled) and the selection rule with its parameters (thresholds, similarity cut-offs, localisation source, decoy generator)	Different rules change task difficulty and what a model can learn	[9, 51, 52]
3	Order of operations: whether negatives were built after the split, and which data each rule used	Using test information for training, preprocessing fits or model selection, when the application would not have it, inflates estimates; building the test set itself necessarily uses evaluation data	[14, 30, 31]
4	The negative rule for training and for testing, and whether they are the same	A shared rule can reward a shortcut that will not exist in use	[9, 35]

#	What to report	Why it matters	Evidence
5	Negative-to-positive ratio in training and test sets, with sensitivity analysis if the true ratio is unknown	Precision-based metrics depend on the ratio	[50, 67]
6	Split type (C1–C3, drug/target settings, cluster, time) and similarity thresholds	Determines which kind of new pair the score describes	[6, 62, 65]
7	How often each entity appears among positives versus negatives	Detects degree imbalance	[13, 36, 45]
8	Results of degree-only and similarity-only baselines	Shows how much performance a shortcut alone explains	[12, 50]
9	AUPRC with the class ratio, AUROC, and PU-corrected estimates with the labelling assumption stated if a class prior is assumed	Makes metrics comparable and honest about imbalance	[58, 67]
10	Variability across negative draws; negative pairs released with stable identifiers (UniProt, ChEMBL or PubChem)	Allows the comparison to be reproduced and re-analysed	[28, 36]

5.5 For groups generating data

Experimental groups can improve the reuse of negative results by retaining the following information:

- **Record the tested space.** Report which pairs were tested, not only which were detected. Without the tested space, a non-detection cannot be told apart from a pair never assayed [7, 40].
- **Report the assay context.** Give the assay type and conditions, the concentration range or activity threshold, the detection limit and the number of replicates. This lets others tell a weak binder from an inactive or an untested pair [41].
- **Deposit inactive results with stable identifiers.** Report inactives alongside actives. Large initiatives now plan to release both positive and negative binding data openly, for example from affinity-selection mass spectrometry and DNA-encoded library screens [71].

6 Negative sampling in other link-prediction settings

PPI prediction is link prediction in a protein network, and DTI prediction is link prediction in a bipartite network of compounds and targets. Other tasks that learn from observed links face the same lack of recorded negatives. They include recommendation from implicit feedback, where

only actions such as clicks or purchases are recorded and dislikes are not. They also include knowledge-graph completion, which predicts missing facts stored as (head, relation, tail) triples, and link prediction in social and information networks. Negative sampling has been reviewed for recommendation [33] and for knowledge-graph representation learning [34]. This section does not survey those fields; it asks which conclusions of the present review carry over to them. Table 4 compares the four settings on three questions: what an unobserved pair can mean, which negatives are used for training and for evaluation, and what information evaluation may use.

Training practice is similar across these fields. Bayesian personalized ranking learns from pairs of an observed and an unobserved item for each user and samples such pairs during training [72]. TransE, a knowledge-graph embedding model, is trained against corrupted triples in which the head or tail entity of a true triple is replaced by a random entity [73]. Both are forms of random sampling of unobserved pairs (Section 3.1), and both accept that some sampled negatives are unrecorded positives. Contrastive representation learning, which learns embeddings by pulling related samples together and pushing negatives apart, makes the same choice when randomly drawn samples serve as negatives. Chuang et al. [74] corrected the contrastive objective for negatives that share the anchor's label, without knowing the true labels; the correction parallels the PU methods of Section 3.7.

The evaluation problems are also shared. Yang et al. [75] showed that common practices in evaluating link prediction can lead to questionable conclusions and, because of extreme class imbalance, recommended precision–recall curves over ROC curves, in line with Section 4.4. Li et al. [76] identified easy negative samples as one of the main pitfalls in benchmarking graph neural networks for link prediction, and proposed an evaluation setting with hard negatives sampled by heuristics. On dynamic graphs, a baseline that only memorises past edges performed surprisingly well because easy negative edges were often used for evaluation, and harder negative-sampling strategies were introduced in response [77]. As with degree-matched negatives in PPI benchmarks (Section 3.4), harder test negatives change the question a benchmark answers, and we expect them to be most informative when reported alongside negatives that represent the candidate pairs. The problem of sampled ranking metrics discussed in Section 4.4 comes from the same literature [70].

Unrecorded positives are handled in different ways. Knowledge-graph completion commonly reports a "filtered" setting, in which corrupted triples that appear in the training, validation or test data are removed before the true triple is ranked [73]. This removes known positives from the negatives but not positives that were never recorded, which is the situation of unlabelled pairs in Section 2.1. In recommendation, Steck [78] treated the absence of ratings as informative and proposed test measures that can be estimated without bias, under mild assumptions, even when ratings are missing not at random. Ratings are missing not at random when whether a rating is observed depends on the rating itself. Missingness in interaction data is not random either: whether an interaction is recorded depends on how often its members have been studied (Sections 2.1 and 3.7). More generally, a negative set chosen by a rule is a sample whose selection depends on properties of the pairs. Zadrozny [79] formalised this sample selection bias for classifier learning, analysed how common learners are affected and proposed a correction that is particularly useful for evaluation.

The data-mining literature also offers a general account of leakage. Kaufman et al. [80] defined leakage as the introduction of information about the target that should not legitimately be available, and showed that it can be avoided by careful data management followed by a "learn-predict separation". Building negatives after the split, from the information each partition is allowed to use (Section 4.3), applies this separation to negative construction. Construction bias (Section 4.2) is a separate problem, because it can arise without any leakage.

Three conclusions appear to carry over. First, negatives chosen to help training need not represent the candidates a model will rank in use, so training and evaluation negatives should be described separately (Section 4.1). Second, easy test negatives can inflate performance, as shown for interaction prediction (Section 3.3) and for static and dynamic graphs [76, 77]. Simple baselines, such as degree-only models (Section 5.3) and memorisation of past edges, show how much of a score such negatives explain. Third, a rule that draws on network statistics or known positives should state which data it uses. In interaction prediction, such rules must not use evaluation labels and usually take their statistics from the training graph (Section 4.3). In knowledge-graph completion, the filtered setting uses known triples from all splits, but only to remove them from the candidates [73]. Other conclusions depend on the task. Whether degree or popularity is a shortcut or a legitimate signal depends on the intended use, as Section 3.1 argues for degree in PPI prediction. Measured negatives with assay thresholds exist only for some interaction tasks

(Section 3.6), where they allow a sampling rule to be checked against pairs that were actually tested. With implicit feedback or an incomplete knowledge graph, negatives come from unobserved pairs, so evaluation depends on assumptions about why data are missing [78]. Finally, classifying test pairs by whether both, one or neither entity appears in training (Section 4.3) corresponds to transductive and inductive evaluation in graph learning [66]. It also corresponds to warm- and cold-start settings in recommendation, in which users or items have or lack an interaction history. Reporting results for each class would make benchmarks easier to compare across these fields.

Table 4. Negatives in four link-prediction settings. Entries summarise the studies cited and are not exhaustive.

Setting	An unobserved pair may be	Training negatives	Evaluation negatives and known problems	Information evaluation may use
PPI and DTI prediction	Untested; tested but not detected, or inactive under an activity threshold; a true interaction not yet recorded (Sections 2.1, 3.6)	Random unobserved pairs; filtered, balanced or property-matched pairs; measured inactives where available (Section 3)	Random, degree-matched or decoy negatives; filters and decoys can make the test easier than the intended task (Sections 3.2–3.5); measured inactives allow a check against tested pairs	Entity-aware splits (C1–C3), with negatives built after the split; network-based rules must not use evaluation labels (Section 4.3)
Recommendation from implicit feedback	An item not seen, seen but not chosen, or a missing positive [33]; data may be missing not at random [78]	Uniformly sampled unobserved items [72]; harder or model-based negatives [33]	Metrics computed on sampled candidates can rank methods differently from full ranking; full ranking or corrected metrics avoid this [70]; measures that allow for data missing not at random [78]	Each user's remaining interactions; in leave-one-out evaluation, one held-out interaction per user is ranked against the items the user has not interacted with [72]
Knowledge-graph completion	A false triple, or a true triple not recorded in the graph [73]	Corrupted triples with a random head or tail [73]; harder sampling methods [34]	True triples among the corrupted candidates can be ranked above the test triple; the filtered setting removes those already known [73]	Known triples from all splits, used only to remove false negatives from the candidate list [73]
Link prediction in static and dynamic graphs	An absent edge, or an edge not yet formed [77]	Random non-edges [77]	Random negatives are easy; heuristic hard negatives [76]; historical and inductive negatives for dynamic graphs [77]; extreme class imbalance favours precision–recall curves [75]	Transductive or inductive setting [66]; chronological splits in dynamic graphs [77]

7 Open problems

In the evaluation by Bernett et al. [12], sequence-based PPI models performed close to random under strict splits that minimised similarity to training proteins. The authors called for further experimental work on poorly characterised interactions. Changing the negative set may reveal these limitations, but cannot be assumed to resolve them.

Uneven study of proteins also complicates the interpretation of unlabelled pairs. The number of recorded partners follows screening frequency [42], affecting the prevalence of hidden positives, the plausibility of PU labelling assumptions and the interpretation of degree-based predictions. These connections motivate methods that model the labelling process explicitly [26].

More extensive negative measurements would help address some of these problems. Systematic screens with a defined tested space [1, 7] and screening campaigns that retain inactive results [39] provide direct sources of such data. Planned open releases of positive and negative binding data may expand the evidence available for DTI models [71]. For PPIs, the Negatome curators noted in 2014 that an experimental method for detecting non-interacting pairs at proteome scale had yet to be invented [3]. This observation highlights a limitation of obtaining negative evidence at scale.

Benchmark design can make the effects of negative construction easier to examine. Treating the selection rule as an explicit parameter and evaluating models under several rules would help distinguish differences between predictors from differences between their test sets [14]. Interpretation of corrected performance estimates remains more demanding: PU corrections for ROC and precision–recall measures depend on a class prior and a labelling assumption [58, 59], and reliable estimation of these priors in interaction data remains an open problem.

Prospective experimental validation remains necessary to establish whether predictions identify interactions in practice [55]. Evaluation on constructed negatives provides useful comparative evidence, but its relevance depends on the candidate pairs and experimental setting to which the model will be applied.

8 Conclusions

Negative construction needs to be considered when interpreting both a model's performance and its advantage over other methods. The studies reviewed here do not support a single strategy for

every PPI or DTI task. Filters, balancing rules and decoys address different problems, and their benefits during training need not carry over to evaluation. Stating the purpose of each negative set helps resolve some apparent disagreements in the literature. For practical comparisons, the essential information is the evidence behind the labels, the pair-selection procedure, the data split and the class ratio. Degree- and similarity-based baselines then help establish what the predictor adds. The same distinctions apply to other link-prediction tasks with incomplete observations, such as recommendation and knowledge-graph completion, although the meaning of a missing link and the availability of measured negatives differ between them (Section 6). Together, these details allow readers to judge whether a reported gain is relevant to the biological application for which the model is proposed.

Statements and Declarations

Funding: This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Competing interests: The author has no relevant financial or non-financial interests to disclose.

Author contributions: As sole author, Justin Li was responsible for conceptualization, investigation, visualization, and writing (original draft, review and editing).

Data availability: No new data were generated or analysed in this review.

Ethics approval: Not applicable.

Use of generative AI: During the preparation of this manuscript, the author used Claude Opus 5 (Anthropic) to improve the language, clarity and readability of the text. The author is responsible for reviewing and verifying the AI-assisted edits and takes full responsibility for the content of the manuscript.

References

1. Luck K, Kim DK, Lambourne L et al (2020) A reference map of the human binary protein interactome. Nature 580:402–408. https://doi.org/10.1038/s41586-020-2188-x
2. Smialowski P, Pagel P, Wong P et al (2010) The Negatome database: a reference set of non-interacting protein pairs. Nucleic Acids Res 38:D540–D544. https://doi.org/10.1093/nar/gkp1026

3. Blohm P, Frishman G, Smialowski P, Goebels F, Wachinger B, Ruepp A, Frishman D (2014) Negatome 2.0: a database of non-interacting proteins derived by literature mining, manual annotation and protein structure analysis. Nucleic Acids Res 42:D396–D400. https://doi.org/10.1093/nar/gkt1079

4. Gilson MK, Liu T, Baitaluk M, Nicola G, Hwang L, Chong J (2016) BindingDB in 2015: A public database for medicinal chemistry, computational chemistry and systems pharmacology. Nucleic Acids Res 44:D1045–D1053. https://doi.org/10.1093/nar/gkv1072

5. Zdrazil B, Felix E, Hunter F et al (2024) The ChEMBL Database in 2023: a drug discovery platform spanning multiple bioactivity data types and time periods. Nucleic Acids Res 52:D1180–D1192. https://doi.org/10.1093/nar/gkad1004

6. Pahikkala T, Airola A, Pietilä S, Shakyawar S, Szwajda A, Tang J, Aittokallio T (2015) Toward more realistic drug-target interaction predictions. Brief Bioinform 16:325–337. https://doi.org/10.1093/bib/bbu010

7. Venkatesan K, Rual JF, Vazquez A et al (2009) An empirical framework for binary interactome mapping. Nat Methods 6:83–90. https://doi.org/10.1038/nmeth.1280

8. Hart GT, Ramani AK, Marcotte EM (2006) How complete are current yeast and human protein-interaction networks? Genome Biol 7:120. https://doi.org/10.1186/gb-2006-7-11-120

9. Neumann D, Roy S, Minhas FUAA, Ben-Hur A (2022) On the choice of negative examples for prediction of host-pathogen protein interactions. Front Bioinform 2:1083292. https://doi.org/10.3389/fbinf.2022.1083292

10. Yaseen A, Amin I, Akhter N, Ben-Hur A, Minhas F (2022) Insights into performance evaluation of compound-protein interaction prediction methods. Bioinformatics 38:ii75–ii81. https://doi.org/10.1093/bioinformatics/btac496

11. Chatterjee A, Walters R, Shafi Z, Ahmed OS, Sebek M, Gysi D, Yu R, Eliassi-Rad T, Barabási AL, Menichetti G (2023) Improving the generalizability of protein-ligand binding predictions with AI-Bind. Nat Commun 14:1989. https://doi.org/10.1038/s41467-023-37572-z

12. Bernett J, Blumenthal DB, List M (2024) Cracking the black box of deep sequence-based protein-protein interaction prediction. Brief Bioinform 25:bbae076. https://doi.org/10.1093/bib/bbae076

13. Li P, Shao B, Zhao G, Liu ZP (2025) Negative sampling strategies impact the prediction of scale-free biomolecular network interactions with machine learning. BMC Biol 23:123. https://doi.org/10.1186/s12915-025-02231-w

14. Cabrera-Andrade A, Robalino-Díaz J, Dueñas-Espín I, Zambrano AK, Villegas-Ch W (2026) BioGraphBench-PPI enables audit-first evaluation of negative-sampling contracts and structural baselines in protein interaction networks. Front Bioinform 6:1966469. https://doi.org/10.3389/fbinf.2026.1966469

15. Jansen R, Gerstein M (2004) Analyzing protein function on a genomic scale: the importance of gold-standard positives and negatives for network prediction. Curr Opin Microbiol 7:535–545. https://doi.org/10.1016/j.mib.2004.08.012

16. Hu L, Wang X, Huang YA, Hu P, You ZH (2021) A survey on computational models for predicting protein-protein interactions. Brief Bioinform 22:bbab036. https://doi.org/10.1093/bib/bbab036

17. Soleymani F, Paquet E, Viktor H, Michalowski W, Spinello D (2022) Protein-protein interaction prediction with deep learning: A comprehensive review. Comput Struct Biotechnol J 20:5316–5341. https://doi.org/10.1016/j.csbj.2022.08.070

18. Chen X, Yan CC, Zhang X, Zhang X, Dai F, Yin J, Zhang Y (2016) Drug-target interaction prediction: databases, web servers and computational models. Brief Bioinform 17:696–712. https://doi.org/10.1093/bib/bbv066

19. Cheng T, Hao M, Takeda T, Bryant SH, Wang Y (2017) Large-Scale Prediction of Drug-Target Interaction: a Data-Centric Review. AAPS J 19:1264–1275. https://doi.org/10.1208/s12248-017-0092-6

20. Ezzat A, Wu M, Li XL, Kwoh CK (2019) Computational prediction of drug-target interactions using chemogenomic approaches: an empirical survey. Brief Bioinform 20:1337–1357. https://doi.org/10.1093/bib/bby002

21. Bagherian M, Sabeti E, Wang K, Sartor MA, Nikolovska-Coleska Z, Najarian K (2021) Machine learning approaches and databases for prediction of drug-target interaction: a survey paper. Brief Bioinform 22:247–269. https://doi.org/10.1093/bib/bbz157

22. Atas Guvenilir H, Doğan T (2023) How to approach machine learning-based prediction of drug/compound-target interactions. J Cheminform 15:16. https://doi.org/10.1186/s13321-023-00689-w

23. Lagarde N, Zagury JF, Montes M (2015) Benchmarking Data Sets for the Evaluation of Virtual Ligand Screening Methods: Review and Perspectives. J Chem Inf Model 55:1297–1307. https://doi.org/10.1021/acs.jcim.5b00090

24. Réau M, Langenfeld F, Zagury JF, Lagarde N, Montes M (2018) Decoys Selection in Benchmarking Datasets: Overview and Perspectives. Front Pharmacol 9:11. https://doi.org/10.3389/fphar.2018.00011

25. Kittelson KS, Martins ACF, Santos RP, Celante G, da Silva Gomes R (2026) Reproducibility, validation, and failure modes across classical and AI-driven molecular docking. J Comput Aided Mol Des 40:137. https://doi.org/10.1007/s10822-026-00849-8

26. Bekker J, Davis J (2020) Learning from positive and unlabeled data: a survey. Mach Learn 109:719–760. https://doi.org/10.1007/s10994-020-05877-5

27. Li F, Dong S, Leier A et al (2022) Positive-unlabeled learning in bioinformatics and computational biology: a brief review. Brief Bioinform 23:bbab461. https://doi.org/10.1093/bib/bbab461

28. Walsh I, Fishman D, Garcia-Gasulla D, Titma T, Pollastri G, ELIXIR Machine Learning Focus Group, Harrow J, Psomopoulos FE, Tosatto SCE (2021) DOME: recommendations for supervised machine learning validation in biology. Nat Methods 18:1122–1127. https://doi.org/10.1038/s41592-021-01205-4

29. Whalen S, Schreiber J, Noble WS, Pollard KS (2022) Navigating the pitfalls of applying machine learning in genomics. Nat Rev Genet 23:169–181. https://doi.org/10.1038/s41576-021-00434-9

30. Kapoor S, Narayanan A (2023) Leakage and the reproducibility crisis in machine-learning-based science. Patterns 4:100804. https://doi.org/10.1016/j.patter.2023.100804

31. Bernett J, Blumenthal DB, Grimm DG, Haselbeck F, Joeres R, Kalinina OV, List M (2024) Guiding questions to avoid data leakage in biological machine learning applications. Nat Methods 21:1444–1453. https://doi.org/10.1038/s41592-024-02362-y

32. Lannelongue L, Inouye M (2024) Pitfalls of machine learning models for protein-protein interaction networks. Bioinformatics 40:btae012. https://doi.org/10.1093/bioinformatics/btae012

33. Ma H, Xie R, Meng L, Feng F, Du X, Sun X, Kang Z, Meng X (2026) Negative sampling in recommendation: a survey and future directions. ACM Trans Inf Syst 44(4):1–44. https://doi.org/10.1145/3793855

34. Madushanka T, Ichise R (2024) Negative sampling in knowledge graph representation learning: a review. arXiv:2402.19195. https://doi.org/10.48550/arXiv.2402.19195

35. Park Y, Marcotte EM (2011) Revisiting the negative example sampling problem for predicting protein-protein interactions. Bioinformatics 27:3024–3028. https://doi.org/10.1093/bioinformatics/btr514

36. Najm M, Azencott CA, Playe B, Stoven V (2021) Drug Target Identification with Machine Learning: How to Choose Negative Examples. Int J Mol Sci 22:5118. https://doi.org/10.3390/ijms22105118

37. Davis MI, Hunt JP, Herrgard S, Ciceri P, Wodicka LM, Pallares G, Hocker M, Treiber DK, Zarrinkar PP (2011) Comprehensive analysis of kinase inhibitor selectivity. Nat Biotechnol 29:1046–1051. https://doi.org/10.1038/nbt.1990

38. Butkiewicz M, Lowe EW, Mueller R, Mendenhall JL, Teixeira PL, Weaver CD, Meiler J (2013) Benchmarking ligand-based virtual High-Throughput Screening with the PubChem database. Molecules 18:735–756. https://doi.org/10.3390/molecules18010735

39. Tran-Nguyen VK, Jacquemard C, Rognan D (2020) LIT-PCBA: An Unbiased Data Set for Machine Learning and Virtual Screening. J Chem Inf Model 60:4263–4273. https://doi.org/10.1021/acs.jcim.0c00155

40. Trabuco LG, Betts MJ, Russell RB (2012) Negative protein-protein interaction datasets derived from large-scale two-hybrid experiments. Methods 58:343–348. https://doi.org/10.1016/j.ymeth.2012.07.028

41. Tang J, Szwajda A, Shakyawar S, Xu T, Hintsanen P, Wennerberg K, Aittokallio T (2014) Making sense of large-scale kinase inhibitor bioactivity data sets: a comparative and integrative analysis. J Chem Inf Model 54:735–743. https://doi.org/10.1021/ci400709d

42. Schaefer MH, Serrano L, Andrade-Navarro MA (2015) Correcting for the study bias associated with protein-protein interaction measurements reveals differences between protein degree distributions from different cancer types. Front Genet 6:260. https://doi.org/10.3389/fgene.2015.00260

43. Liu H, Sun J, Guan J, Zheng J, Zhou S (2015) Improving compound-protein interaction prediction by building up highly credible negative samples. Bioinformatics 31:i221–i229. https://doi.org/10.1093/bioinformatics/btv256

44. Eid FE, ElHefnawi M, Heath LS (2016) DeNovo: virus-host sequence-based protein-protein interaction prediction. Bioinformatics 32:1144–1150. https://doi.org/10.1093/bioinformatics/btv737

45. Yu J, Guo M, Needham CJ, Huang Y, Cai L, Westhead DR (2010) Simple sequence-based kernels do not predict protein-protein interactions. Bioinformatics 26:2610–2614. https://doi.org/10.1093/bioinformatics/btq483

46. Huang N, Shoichet BK, Irwin JJ (2006) Benchmarking sets for molecular docking. J Med Chem 49:6789–6801. https://doi.org/10.1021/jm0608356

47. Mysinger MM, Carchia M, Irwin JJ, Shoichet BK (2012) Directory of useful decoys, enhanced (DUD-E): better ligands and decoys for better benchmarking. J Med Chem 55:6582–6594. https://doi.org/10.1021/jm300687e

48. Guo Y, Yu L, Wen Z, Li M (2008) Using support vector machine combined with auto covariance to predict protein-protein interactions from protein sequences. Nucleic Acids Res 36:3025–3030. https://doi.org/10.1093/nar/gkn159

49. Peng L, Zhu W, Liao B, Duan Y, Chen M, Chen Y, Yang J (2017) Screening drug-target interactions with positive-unlabeled learning. Sci Rep 7:8087. https://doi.org/10.1038/s41598-017-08079-7

50. Dunham B, Ganapathiraju MK (2022) Benchmark Evaluation of Protein-Protein Interaction Prediction Algorithms. Molecules 27:41. https://doi.org/10.3390/molecules27010041

51. Ben-Hur A, Noble WS (2006) Choosing negative examples for the prediction of protein-protein interactions. BMC Bioinformatics 7(Suppl 1):S2. https://doi.org/10.1186/1471-2105-7-S1-S2

52. Chen L, Cruz A, Ramsey S, Dickson CJ, Duca JS, Hornak V, Koes DR, Kurtzman T (2019) Hidden bias in the DUD-E dataset leads to misleading performance of deep learning in structure-based virtual screening. PLoS One 14:e0220113. https://doi.org/10.1371/journal.pone.0220113

53. Sieg J, Flachsenberg F, Rarey M (2019) In Need of Bias Control: Evaluating Chemical Data for Machine Learning in Structure-Based Virtual Screening. J Chem Inf Model 59:947–961. https://doi.org/10.1021/acs.jcim.8b00712

54. Wallach I, Heifets A (2018) Most Ligand-Based Classification Benchmarks Reward Memorization Rather than Generalization. J Chem Inf Model 58:916–932. https://doi.org/10.1021/acs.jcim.7b00403

55. Cichonska A, Ravikumar B, Parri E, Timonen S, Pahikkala T, Airola A, Wennerberg K, Rousu J, Aittokallio T (2017) Computational-experimental approach to drug-target interaction mapping: A case study on kinase inhibitors. PLoS Comput Biol 13:e1005678. https://doi.org/10.1371/journal.pcbi.1005678

56. Elkan C, Noto K (2008) Learning classifiers from only positive and unlabeled data. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, New York, pp 213–220. https://doi.org/10.1145/1401890.1401920

57. Mordelet F, Vert J-P (2014) A bagging SVM to learn from positive and unlabeled examples. Pattern Recognit Lett 37:201–209. https://doi.org/10.1016/j.patrec.2013.06.010

58. Jain S, White M, Radivojac P (2017) Recovering True Classifier Performance in Positive-Unlabeled Learning. In: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence. AAAI Press, Palo Alto. https://doi.org/10.1609/aaai.v31i1.10937

59. Ramola R, Jain S, Radivojac P (2019) Estimating classification accuracy in positive-unlabeled learning: characterization and correction strategies. Pac Symp Biocomput 24:124–135

60. Hamp T, Rost B (2015) More challenges for machine-learning protein interactions. Bioinformatics 31:1521–1525. https://doi.org/10.1093/bioinformatics/btu857

61. Geirhos R, Jacobsen J-H, Michaelis C, Zemel R, Brendel W, Bethge M, Wichmann FA (2020) Shortcut learning in deep neural networks. Nat Mach Intell 2:665–673. https://doi.org/10.1038/s42256-020-00257-z

62. Park Y, Marcotte EM (2012) Flaws in evaluation schemes for pair-input computational predictions. Nat Methods 9:1134–1136. https://doi.org/10.1038/nmeth.2259

63. Szymborski J, Emad A (2022) RAPPPID: towards generalizable protein interaction prediction with AWD-LSTM twin networks. Bioinformatics 38:3958–3967. https://doi.org/10.1093/bioinformatics/btac429

64. Mayr A, Klambauer G, Unterthiner T, Steijaert M, Wegner JK, Ceulemans H, Clevert DA, Hochreiter S (2018) Large-scale comparison of machine learning methods for drug target prediction on ChEMBL. Chem Sci 9:5441–5451. https://doi.org/10.1039/c8sc00148k

65. Sheridan RP (2013) Time-split cross-validation as a method for estimating the goodness of prospective prediction. J Chem Inf Model 53:783–790. https://doi.org/10.1021/ci400084k

66. Hamilton WL, Ying R, Leskovec J (2017) Inductive representation learning on large graphs. In: Advances in Neural Information Processing Systems 30. Curran Associates, Red Hook. https://proceedings.neurips.cc/paper_files/paper/2017/hash/5dd9db5e033da9c6fb5ba83c7a7ebea9-Abstract.html

67. Saito T, Rehmsmeier M (2015) The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. PLoS One 10:e0118432. https://doi.org/10.1371/journal.pone.0118432

68. Davis J, Goadrich M (2006) The relationship between Precision-Recall and ROC curves. In: Proceedings of the 23rd International Conference on Machine Learning. ACM, New York, pp 233–240. https://doi.org/10.1145/1143844.1143874

69. Sledzieski S, Singh R, Cowen L, Berger B (2021) D-SCRIPT translates genome to phenome with sequence-based, structure-aware, genome-scale predictions of protein-protein interactions. Cell Syst 12:969–982.e6. https://doi.org/10.1016/j.cels.2021.08.010

70. Krichene W, Rendle S (2020) On Sampled Metrics for Item Recommendation. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, New York, pp 1748–1757. https://doi.org/10.1145/3394486.3403226

71. Edwards AM, Owen DR, Structural Genomics Consortium Target 2035 Working Group (2025) Protein-ligand data at scale to support machine learning. Nat Rev Chem 9:634–645. https://doi.org/10.1038/s41570-025-00737-z

72. Rendle S, Freudenthaler C, Gantner Z, Schmidt-Thieme L (2009) BPR: Bayesian personalized ranking from implicit feedback. In: Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence. AUAI Press, Arlington, pp 452–461

73. Bordes A, Usunier N, Garcia-Duran A, Weston J, Yakhnenko O (2013) Translating embeddings for modeling multi-relational data. In: Advances in Neural Information Processing Systems 26. Curran Associates, Red Hook. https://proceedings.neurips.cc/paper_files/paper/2013/hash/1cecc7a77928ca8133fa24680a88d2f9-Abstract.html

74. Chuang C-Y, Robinson J, Lin Y-C, Torralba A, Jegelka S (2020) Debiased contrastive learning. In: Advances in Neural Information Processing Systems 33. Curran Associates, Red Hook, pp 8765–8775. https://proceedings.neurips.cc/paper_files/paper/2020/hash/63c3ddcc7b23daa1e42dc41f9a44a873-Abstract.html

75. Yang Y, Lichtenwalter RN, Chawla NV (2015) Evaluating link prediction methods. Knowl Inf Syst 45:751–782. https://doi.org/10.1007/s10115-014-0789-0

76. Li J, Shomer H, Mao H, Zeng S, Ma Y, Shah N, Tang J, Yin D (2023) Evaluating graph neural networks for link prediction: current pitfalls and new benchmarking. In: Advances in Neural Information Processing Systems 36. Curran Associates, Red Hook, pp 3853–3866. https://proceedings.neurips.cc/paper_files/paper/2023/hash/0be50b4590f1c5fdf4c8feddd63c4f67-Abstract-Datasets_and_Benchmarks.html

77. Poursafaei F, Huang S, Pelrine K, Rabbany R (2022) Towards better evaluation for dynamic link prediction. In: Advances in Neural Information Processing Systems 35. Curran Associates, Red Hook, pp 32928–32941. https://proceedings.neurips.cc/paper_files/paper/2022/hash/d49042a5d49818711c401d34172f9900-Abstract-Datasets_and_Benchmarks.html

78. Steck H (2010) Training and testing of recommender systems on data missing not at random. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, New York, pp 713–722. https://doi.org/10.1145/1835804.1835895

79. Zadrozny B (2004) Learning and evaluating classifiers under sample selection bias. In: Proceedings of the 21st International Conference on Machine Learning. ACM, New York, p 114. https://doi.org/10.1145/1015330.1015425

80. Kaufman S, Rosset S, Perlich C, Stitelman O (2012) Leakage in data mining: formulation, detection, and avoidance. ACM Trans Knowl Discov Data 6(4):15. https://doi.org/10.1145/2382577.2382579