On the Unreported-Profile-is-Negative Assumption for Predictive Cheminformatics

Chao Lan, Sai Nivedita Chandrasekaran, and Jun Huan

Abstract—In cheminformatics, compound-target binding profiles have been a main source of data for research. However, for data repositories that only provide positive profiles, a popular assumption is that unreported profiles are all negative. In this paper, we caution the audience not to take this assumption for granted, and present empirical evidence of its ineffectiveness from a machine learning perspective. Our examination is based on a setting where binding profiles are used as features to train predictive models; we show (1) prediction performance degrades when the assumption fails and (2) explicit recovery of unreported profiles improves prediction performance. In particular, we propose a framework that jointly recovers profiles and learns predictive model, and show it achieves further performance improvement. The presented study not only suggests applying matrix recovery methods to recover unreported profiles, but also initiates a new missing feature problem which we called Learning with Positive and Unknown Features.

Index Terms—Chemical-target interaction, GPCR, missing data, collaborative filtering, matrix factorization

1 INTRODUCTION

In cheminformatics, compound-target binding profiles have been a main source of data for research. The profile is usually binarized, in a way that positive profile between a compound and a target indicates they interact while negative profile between them indicates otherwise. When data repositories only provide positive binding profiles, a common assumption is that all unreported profiles are negative. In a recent study [1], authors pointed out this assumption may be potentially wrong, but continued to adopt it and argued that unreported profiles may be limited in number and thus have limited impact on analysis. But how would such unreported-profile-is-negative assumption affect analysis? And what remedy could be done to mitigate its effect? Little efforts have been made to address these questions, and we therefore caution audience not to take the assumption for granted. Collecting direct evidence to verify the above assumption would be impractical, since examining all unreported files through chemical experiments can be expensive— it also unnecesitate the assumption as all unreported profiles are not reported. However, we believe it is practical to collect indirect evidence, such as how methods will perform based on this assumption, as compared to the methods that are not based on this assumption. This motivates our study. In this paper, we investigate the unreported-profile-is-negative assumption from a machine learning perspective, based on a setting where binding profiles are used as features to train predictive models for predicting a compound’s interactions with new targets. Our experimental results show that prediction performance degrades when the assumption fails, and explicit recovery of unreported profiles (using one-class matrix recovery methods e.g., [2]) improves prediction performance. A straightforward approach is to first apply profile recovery and then apply predictive learning. We hypothesize this can be further improved by integrating both phases. To this end, we propose a framework that jointly optimize profile recovery and predictive learning, and show it improves performance on synthetic data set (with ground truth) and the real-world Laboratory of Molecular Modeling and Design data repository [3] (without ground truth).

1.1 A New Missing Data Problem for Machine Learning

From a technical perspective, a major contribution of this paper is the introduction of a new machine learning problem which we called Learning with Positive and Unknown Features.

The problem considers supervised learning with binary featured data. It assumes one feature value is completely missing in the training sample, while the other feature value is partly missing. This setting corresponds to the cheminformatics task that uses binding profiles as features in predictive learning, where negative profiles are completely missing and positive files are partly missing (i.e., partly reported in the data repository), as illustrated in Fig. 1.

This problem is related to three machine learning problems, namely, (a) learning with missing data, (b) positive and unlabeled learning and (c) one-class collaborative filtering. In the following, we will briefly introduce each problem and discuss their connection and difference to ours.

The problem we introduced is related to but essentially different from three machine learning topics, namely, (a) learning with missing data, (b) positive and unlabeled learning and (c) one-class collaborative filtering. Learning with missing data is an aged problem (e.g., [4]), which includes learning with missing features. However, prior solutions mostly assume all feature values are partly
missing, while we assume one feature value is completely missing. This difference could break down prior solutions in fundamental ways. As an example, consider the popular nearest-neighbor based imputation approach, which replaces a missing value with the average of its neighboring observed values. In our problem Fig. 1, this approach will replace all missing values with 1', resulting a sample matrix of all 1’s.

Positive and unlabeled learning (e.g., [5], [6]) refers to the problem of learning with binary label set, and only positive examples are observed in the training sample while negative examples are completely missing. This problem assumes the same missing pattern as ours, but on different aspects: they assume labels are missing while we assume features are missing. So their solutions do not directly apply to our problem.

Perhaps the most related topic is one-class collaborative filtering (o.c.c.f.) (e.g., [2], [7]), which refers to the problem of recovering an incomplete matrix where one entry value is partly missing while the rest entry values are completely missing. In our problem, the incomplete feature matrix is a one-class matrix, hence existing solutions to o.c.c.f. can be directly applied. However, the two problems have different goals: o.c.c.f. only aims to recover missing features, while we aim to learn predictive model based on missing features. This seemingly subtle difference can lead to fundamentally different schools of thinking; while accurate feature recovery suffices for learning accurate predictive model, whether it is necessary remains an open question after all, it would not be hard to imagine different sample distributions could admit similar decision boundaries. (It is this thinking that motivates our joint feature recovery and predictive learning framework, allowing the latter to bias the former.)

Each of the above topics has been extensively studied. For audience interested in positive and unlabeled learning and one-class collaborative filtering, we present surveys on both topics at the end of this paper; learning with missing data is a huge topic, and we refer audience to [4].

1.2 Organization

The remainder of this paper is organized as follows: in Section 2, we introduce notations and the basic problem setting; in Section 3, we introduce several prediction approaches, including one that follows the unreported-profile-is-negative assumption, one that first recovers unreported profiles and then performs predictive learning, and one that jointly recovers unreported profiles and performs predictive learning; in Section 4, we present experimental studies and results.

2 Notations and Problem Setting

For a number \( t > 0 \), let \( [t] := \{1, 2, ..., t\} \). For a vector \( V \), let \( V_i \) be its \( i_{th} \) element. For a matrix \( M \), let \( M_{ij} \) be its element at row \( i \) and column \( j \), let \( M_{ij} \) be its \( i_{th} \) row vector and \( M_j \) be its \( j_{th} \) column vector; let \( M' \) be its transpose, \( \text{tr}(M) \) be its trace and \( \|M\| \) be its Frobenius norm (or, \( \ell_2 \) norm if \( M \) is a vector); let \( M' := [M, 1] \) be an augmented matrix of \( M \), where 1 is a properly-sized vector of ones. Let \( I_b \) be a vector of value \( b \) and \( I \) be an identity matrix, both properly sized by the context.

Consider the task of supervised learning. Let there be a labeled training sample consisting of \( n \) examples described by \( p \) binary features defined on \{0, 1\}. We will represent the sample by \((X, Y) \in \mathbb{R}^{n \times p} \times \mathbb{R}^n \), where \( X \) is its feature matrix with \( X_i \) being the feature vector of example \( i \), and \( Y \) is its label vector with \( Y_i \) being the label of example \( i \). For simplicity, assume \( Y \) is binary.

We have proposed the problem of learning with positive and unknown features, as illustrated in Fig. 1. In the above setting, this corresponds to a scenario where feature value 0 is completely missing in \( X \), while feature value 1 is partly missing in \( X \). We do not make statistical assumption on the missing pattern as in traditional missing data study [4]; instead, we make a common matrix recovery assumption that the feature matrix has low rank (e.g., when features are correlated) – it should be mentioned, however, this assumption is only needed to theoretically justify our later application of matrix recovery methods, but in practice these methods work just fine without the assumption being strictly verified e.g., [8].

Just for clarity, our goal is to estimate a predictive function (mapping from the sample set to the label set) based on the incomplete training sample \((X, Y)\).

3 Examined Approaches

In this section, we present several approaches used to examine the effect of unreported-profile-is-negative assumption on prediction performance. These approaches include one that assumes unreported profiles are negative, two that first recover unreported profiles (using mean imputation and o.c.c.f. methods respectively), and our proposed one that jointly recovers unreported profiles and learn predictive models. For fair comparison, all approaches use ridge regression as the base predictive model.

3.1 Direct Supervised Learning (DSL)

This approach adopts the unreported-profile-is-negative assumption. Recall in the feature matrix \( X \), value 0 is completely missing and value 1 is partly missing. The approach simply replaces all missing values in \( X \) with value 0, and then performs standard supervised learning on the updated sample \((X, Y)\); specifically, it estimates a ridge regression vector \( \beta \in \mathbb{R}^{p+1} \) that minimizes

\[
\mathcal{L}_1(\beta) = ||X \beta - Y||^2 + \lambda_1 ||\beta||^2, \tag{1}
\]

discussions and conclusions are presented in Section 5 and in Section 6, respectively.
where $\lambda_1 > 0$ is a regularization coefficient. It is well-known the above problem admits an analytic solution

$$
\beta = (X'X + \lambda I)^{-1} X'Y.
$$

We call this approach Direct Supervised Learning (DSL), and use it as a baseline.

### 3.2 Imputation + Supervised Learning (ISL)

This approach first recovers unreported profiles using a traditional statistical imputation method, then performs standard supervised learning.

In this study, we employ the classic mean imputation method [4], which replaces a missing feature value of an example with the average observed values of the same feature of same-class examples. In our context, this means replacing missing value $X_{ij}$ with

$$
\hat{X}_{ij} = \sigma \left( \frac{1}{N_{ij}} \sum_a X_{aj} \cdot 1_{\{a,j\} \in O, Y_a = Y_i} \right),
$$

where $\sigma(\cdot)$ is a function rounding its input to $\{0, 1\}$, $O$ is the index set of all observed elements in $X$ and

$$
N_{ij} = |\{a \in [n]; (a, j) \in O, Y_a = Y_i \}|
$$

is the number of examples which have feature $j$ observed and belong to the same class as example $i$.

Now, we see a problem if one directly applies the above approach to recovery missing values in $X$: all missing values will eventually be replaced by 1, because only value 1 has been observed; the consequence is no decision boundary can distinguish positive examples and negative examples, since they are squeezed into one point. This limitation applies to most traditional imputation methods, which (implicitly) assume no feature value is completely missing.

To lift the above limitation in application, we first randomly select a $\delta_i$ fraction of missing values and assume they are indeed negative (i.e., replaces them with 0), then apply mean imputation to recover the rest missing values. Similar strategy can be found in [9].

We call this approach Imputation plus Supervised Learning (ISL). It is used to show the benefit of feature recovery against assuming unreported features are negative; it also serves as a baseline feature recovery method to show the benefit of our employed o.c.c.f. feature recovery methods.

### 3.3 OCCF + Supervised Learning (OCSL)

This approach first recovers unreported binding profiles using a modern matrix recovery method called one-class collaborative filtering (o.c.c.f) [2], then performs supervised learning on the recovered sample. There are two methods proposed in [2], and we employ the weighted alternate least square (wALS) method with uniform weighting scheme, which recovers missing values using a weighted low-rank factorization model of $X$. Suppose the rank of $X$ is $r$, i.e., the wALS method seeks for two factors $U \in \mathbb{R}^{n \times r}$ and $V \in \mathbb{R}^{p \times r}$ that minimizes

$$
\mathcal{L}_2(U, V) = W \circ ||X - UV'||^2 + \lambda_2 \cdot \text{tr}(U'D'U + V'D'V),
$$

where $\circ$ is the Hadamard product, $\lambda_2$ is the regularization coefficient and $W \in \mathbb{R}^{n \times p}$ is the observation indicator matrix defined as

$$
W_{ij} = \begin{cases} 
1, & \text{if } X_{ij} \text{ is observed} \\
\delta_{wi}, & \text{if } X_{ij} \text{ is missing}
\end{cases},
$$

where $\delta_{wi}$ is a small constant representing one’s confidence that missing values are 0’s (i.e., unreported profiles are indeed all negative); matrices $D^r \in \mathbb{R}^{n \times n}$ and $D^c \in \mathbb{R}^{p \times p}$ are both diagonal and defined as

$$
D^r_{ii} = \sum_{j \in [p]} W_{ij},
$$

and

$$
D^c_{jj} = \sum_{i \in [n]} W_{ij}.
$$

Minimizing (5) can be solved by alternately updating

$$
U_i = X_i \hat{W}^i V \left( V' \hat{W}^i V + \lambda_2 \sum_j W_{ij} I \right)^{-1},
$$

and

$$
V_j = X_j \hat{W}^j U \left( U' \hat{W}^j U + \lambda_2 \sum_i W_{ij} I \right)^{-1},
$$

where both $\hat{W}^i \in \mathbb{R}^{p \times p}$ and $\hat{W}^j \in \mathbb{R}^{n \times n}$ are diagonal matrices such that

$$
\hat{W}^i_{kk} = W_{ik},
$$

and

$$
\hat{W}^j_{kk} = W_{kj}.
$$

It should be mentioned (5) is a matrix representation of the wALS method we derived to facilitate implement.

We call this approach OCCF plus Supervised Learning (OCSL), and it is the first approach we suggest for the problem of learning with positive and unknown features. Since this approach separates feature recovery and supervised learning, it also serves as a baseline to show the benefit of our next proposed joint feature recovery and supervised learning approach.

### 3.4 Joint OCCF and Supervised Learning (JOCSL)

Both ISL and OCSL approaches separately recover unreported profiles and supervised learning. In this section, we propose to jointly performs both tasks.

The rational behind this joint framework is that, feature recovery and supervised learning may bias each other in a beneficial way, in a sense that even if the recovered features are less accurate, they may still result in more accurate predictive model. (Recall that differently recovered samples may still admit similar decision boundaries, hence similar predictive models.) In a worst case, the introduced bias may help to reduce estimation variance of the predictive model.

To this end, we integrate the objective $\mathcal{L}_1$ of predictive learning and the objective $\mathcal{L}_2$ of feature recovery, and propose to minimize the following joint objective
\[ L_3(U, V, \beta) = W \circ \|X - UV'\|^2 + \alpha \|(UV)_{+} \cdot \beta - Y\|^2 + \lambda_2 \cdot \text{tr}(U'D'U + V'D'V) + \lambda_1 \|\beta\|^2. \]  

(13)

Objective \( L_3 \) can be interpreted as follows: the first term aims to recover unreported profiles, the second term aims to learn predictive model based on the recovered feature matrix \( UV' \). There are three hyper-parameters in \( L_3 \), namely, \( \alpha, \lambda_1 \) and \( \lambda_2 \). Among them, \( \alpha \) is a key hyper-parameter weighting the importance of feature recovery and predictive learning – a large \( \alpha \) is more likely to allow predictive learning to bias feature recovery, while a small \( \lambda \) is more likely to allow feature recovery to bias predictive learning; our general recommendation is to choose a relatively large \( \lambda \) under the hypothesis that accurate feature recovery is not necessary for accurate predictive learning. (Of course, one does not need too large \( \lambda \) that completely ignores the impact of feature recovery on predictive learning.) The other two hyper-parameters \( \lambda_1, \lambda_2 \) are standard regularization coefficients—if one has to construct a big regression model or factorization model but does not have enough labeled examples or reported binding profiles, one may choose larger \( \lambda_1, \lambda_2 \) to more strongly regularize the models.

Our goal is to learn \((U, V, \beta)\) that minimizes formula (13). Similar to prior work, we solve this by alternatingly updating each variable. For any variable \( S \), its update rule is obtained by setting the partial derivative of \( L_3 \) with respect to \( S \) to zero, and solve the equation for \( S \). Derivatives for the first and third terms in (13) can be directly obtained from [2], and their update rules are (9) and (10). We only need to derive derivatives for the second and fourth terms in (13).

Re-write the regression coefficient as \( \beta = [\beta_{[p]}; \beta_{[p+1]}] \), where \( \beta_{[p]} \) is a \( p \)-dimensional vector containing the first \( p \) elements of \( \beta \) and \( \beta_{[p+1]} \) is the last element of \( \beta \) (i.e., the bias term). Write

\[ h := \|(UV)'_+ \cdot \beta - Y\|^2. \]  

(14)

It is easy to verify that

\[ h = \sum_{i \in [n]} \left( \sum_{j \in [p]} (U_i V'_j) \cdot \beta_j + \beta_{[p+1]} - Y_i \right)^2. \]  

(15)

Taking derivatives of \( h \), we have

\[ \frac{\partial h}{\partial U_i} = 2 (U_i V'_j) \cdot \beta_j + \beta_{[p+1]} - Y_i \]  

(16)

\[ \frac{\partial h}{\partial V_j} = 2 \beta_j U' ( (UV)'_+ \cdot \beta - Y), \]  

(17)

and

\[ \frac{\partial h}{\partial \beta} = 2 (UV)'_+ ( (UV)'_+ \cdot \beta - Y). \]  

(18)

Based on (16), (17) and (18), we take derivatives of \( L_3 \) with respect to different variables and set them to zero. Solving for each variable gives the following update rules

\[ U_i = \left( X_i W' V + \alpha(Y_i - \beta_{[p+1]}) \beta_{[p]} V \right) \cdot \left( V' W' V + \lambda_2 \sum_j W_j I + \alpha \lambda V' \beta_{[p]} \beta_{[p]}' V \right)^{-1}, \]  

(19)

\[ V_j = \left( X_i W' j U + 2 \beta_j (Y - 1 \beta_{[p]} Y - \beta_{[p]} U U') \right) \cdot \left( U' W' j U + \lambda_2 \sum_i W_j I + 2 \beta_j^2 U U' \right)^{-1}, \]  

(20)

where \( \beta \), a vector same as \( \beta_{[p]} \) except its \( j \)th entry is zero; the update rule for \( \beta \) is

\[ \beta = \frac{( (UV)'_+ (UV)' + \lambda I)^{-1} (UV)' Y - \beta_{[p]} Y}. \]  

(21)

Based on (19), (20) and (21), we alternately update \( U, V \) and \( \beta \) until proper termination criterion is met. In practice, we first update \( U \) and \( V \) until the feature matrix recovery error (measured by MSE over the entire matrix) converge; then we include \( \beta \) in the loop of update.

We call this approach Joint OCCF and Supervised Learning (JOCSL). It is the second approach we suggest for the problem of learning with positive and unknown features.

4 EXPERIMENTS

In this section, we empirically evaluate prediction performance of the previously introduced approaches, for collecting indirect evidence on the effectiveness of the unreported-profile-is-negative (u.p.n.) assumption. Recall DSL is the only approach assuming all unreported profiles are negative, while other three approaches are not based on this assumption; hence it would be fair to say the assumption is ineffective if DSL consistently performs worse than other approaches. In addition, we also examine the effectiveness of the proposed joint feature recovery and supervised learning framework, by comparing performance of JOCSL and other two approaches ISL and OCSL. We have posted the source code of all implementations.

4.1 Experiments on a Synthetic Dataset

For better understanding the problem of learning with positive and unknown features, and the performance of our introduced approaches, it is desirable to have ground truth of all missing values (i.e., unreported profiles). However, this information is usually difficult, if not impossible, to collect on real-world data sets. We thus first experiment on a synthetic data set.

Our data set was constructed as follows: it contains 200 examples described by 50 binary features \([0, 1]\) and equally divided into a positive class and negative class. All feature values were drawn i.i.d. from some Bernoulli distribution – for the first 10 features of positive examples and the last 10 features of negative examples, we set the probability of taking value 1 as \( p_1 = 0.4 \); for the remaining features, we set this probability as \( p_2 = 0.2 \). Such generation guarantees the two classes are largely separable; their distributions are shown in Fig. 2 (left).

Our first experiment demonstrates how the u.p.n. assumption may affect sample distribution. In the constructed data
set, we hid all feature value 0’s and 25 percent randomly selected feature value 1’s; then, based on the unreported-profile-is-negative assumption, all hidden values were replaced with 0’s. Sample distribution of the recovered data set is shown in Fig. 2 (left), with \( \delta = 0.25 \). Comparing the two sample distributions in Fig. 2, we see the unreported-profile-is-negative assumption pushes two classes towards each other, increasing the difficulty of learning; the recovered distribution also seems distorted, suggesting it may mis-bias the decision boundary.

Our second experiment suggests the unreported-profile-is-negative assumption does increase the difficulty of learning. From the constructed data set, we randomly selected 50 percent examples for training and used the rest for testing. Then, we hid all feature value 0’s and a fraction \( \delta \) of randomly selected feature value 1’s in the feature matrix \( X \), resulting in an incomplete feature matrix \( \hat{X} \). The performance of DSL trained on \((\hat{X}, Y)\) under different \( \delta \) is shown in Fig. 3 (left). We see prediction accuracy generally decreases as \( \delta \) increases, i.e., as more unreported (positive) profiles are mis-assumed negative.

One may wonder whether the accuracy degeneration may be less serious if the data set contains a lot of truly positive profiles. Our third experiment suggests this is unfortunately not the case; our results even suggest the opposite, i.e., the more positive profiles a data set contains, the more accuracy loss one suffers by applying the unreported-profile-is-negative assumption in predictive learning. To this end, we increased the two Bernoulli probabilities \( p_1 \) and \( p_2 \), as they directly determine the number of truly positive profiles in the data set. The performance of DSL is shown in Fig. 3 (right).

Our last experiment evaluated the performance of all approaches on the synthetic data set. The hyper-parameters were \( \lambda_1 = 1, \lambda_2 = 10^{-4}, \alpha = 2, \delta_s = 0.5, \delta_w = 0.25 \) and factorization rank was 20; these values were chosen among a pool of candidate configurations, and we believe they are representative. Results are shown in Fig. 4.

In Fig. 4, we see DSL performs worse than most approaches, implying the unreported-profile-is-negative assumption is ineffectiveness in the prediction task. We also have other observations. First, the four examined approaches degrades as \( \delta \) increases (i.e., more positive profiles are unreported); this is not surprising. Second, OCSC and JOCSC outperform ISL and DSL, suggesting one-class collaborative filtering is an effectiveness method to tackle unreported profiles in predictive learning. Somewhat surprisingly, we see both OCSC and JOCSC perform better than the IDEAL approach when there are limited unreported profiles, which may be because the introduced \( o.c.c.f. \) component has increased the complexity of predictive model, allowing it to better fit the underlying sample distribution. Third, JOCSC performs better than OCSC, suggesting our proposed joint feature recovery and predictive learning framework is effective; this may be because JOCSC can more properly control model complexity, since it has the feature recovery and predictive learning components biasing each other. Finally, the ISL method somehow performs poorly on this data set, which may be because feature values are generated independently (hence it does not make sense for ISL to impute a missing value by mean of observed values).

4.2 Experiment on a Real-World Dataset

In this section, we experiment on a real-world cheminformatics data set. We collected data from the popular Laboratory of Molecular Modeling and Design (LMMD) repository [3]. Our data set contains interaction information between 203 drugs and two sets of proteins, namely, 95 Enzyme proteins and 240 GPCR proteins. The number of drug-Enzyme interactions per Enzyme protein and drug are shown in Fig. 5. We treated each drug as an instance, treated its interactions with Enzyme proteins as features, and its interactions with GPCR proteins as labels. The task is to predict a drug’s interactions with GPCR proteins based on its interactions with Enzyme proteins.

Now, the problem is the LMMD repository only provides positive interaction information, and a common assumption is that unreported interactions are negative. For better understanding this issue, we took a snapshot of the data repository and show it in Table 1. It can be seen drug 10161 interacts with protein D00528, and thus their interaction information will be encoded as 1 in the feature matrix; on the other hand, the interaction between drug 10800 and protein D00528 is not reported, hence their interaction is commonly assumed negative and encoded as 0.

We selected 10 GPCR protein to construct 10 predictive learning tasks: each task use all drug-Enzymes interactions as features and drug interaction with one GPCR protein as label. Since the original data set is highly imbalanced in positive and negative interactions, as illustrated in Fig. 6, we selected these 10 proteins which have the most balanced
interactions; we then down-sampled negative examples in each task so that each has equal positive and negative interactions. Negative examples farthest to positive examples (on average) were selected during the down-sampling process. We evaluated the performance of examined approaches on each task. The experimental protocol is the same as before, and most hyper-parameters are grid searched. All grid-searched results are shown in Figs. 7 and 8, one reporting classification accuracy and the other f1 score.

Fig. 4. Classification accuracy of all approaches on testing data, based on \((X, Y)\) under different \(\delta\); all results are obtained by averaging results of 50 random choices of training examples.

Let us first discuss the classification accuracies in Fig. 7. The DSL approach does not give the best performance in most tasks, suggesting the unreported-profile-is-negative \((u:p:n:\) assumption is ineffectiveness. The JOCSL approach outperforms DSL in most tasks, suggesting its effectiveness in dealing with unreported profiles. The OCSL approach, on the other hand, improves DSL occasionally; this seems to suggest the performance gain of JOCSL is largely due to its joint feature recovery and supervised learning framework. We also see the performance of JOCSL is superior to (and somewhat tied to) the performance of OCSL. The ISL approach improves DSL occasionally and incrementally; we believe its performance is limited by the assumption that both 0 and 1 feature values are (sufficiently) observed in training samples, which is not satisfied in our problem of learning with positive and unknown features.

Why do we examine f1 score in Fig. 8? Even though the constructed tasks are balanced, it is possible some positive drug-GPCR protein interactions are not reported. In our prediction tasks, this means some collected negative labels may not be truly negative but in fact positive. In this case, evaluations based on negative labels may not be very reliable, so we focus on positive labels. The f1 score is

\[
f1 = \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}},
\]

where

\[
\text{precision} = \frac{|\text{predicted & truly positive labels}|}{|\text{predicted positive labels}|}
\]

and

\[
\text{recall} = \frac{|\text{predicted & truly positive labels}|}{|\text{truly positive labels}|}.
\]

In Fig. 8, we see DSL does not give the best f1 scores in most tasks, again, suggesting the \(u:p:n:\) assumption is ineffective. Different from previous experiment, however, we see OCSL outperforms JOCSL occasionally. This suggests OCSL is better at recognizing positive examples, while comparatively JOCSL recognizes positive and negative examples equally well. This may be because JOCSL has biased feature recovery using both positive and negative labels, hence the recovered features would encourage decision boundary to recognize both classes equally well. The implication is as follows: if one trusts the collected labels, JOCSL may be a better choice; if one does not trust the negative labels, OCSL may be a better choice.

Finally, we summarized the best performance of each approach in each task in Table 2 and Table 3 respectively. We see JOCSL gives best prediction accuracy in most tasks, while JOCSL, OCSL and ISL perform equally well under the f1 score measure. All results suggest the \(u:p:n:\) assumption is
ineffective, since DSL is the only approach based on this assumption and performs consistently worse than other approaches.

5 DISCUSSIONS

This paper aims to caution researchers not to take the unreported-profile-is-negative assumption for granted in cheminformatics research. We present empirical evidence of its deficiency from a machine learning perspective, such that a standard supervised learning approach based on this assumption can perform worse than approaches which are not based on this assumption and recover unreported profiles using the o.e.c.f method.

This paper considers a supervised learning setting, where binding profiles of one compound family are directly used as features to train models for predicting profiles of another compound family. The biological rational is inherited from

| Approach | Task 1 | Task 2 | Task 3 | Task 4 | Task 5 | Task 6 | Task 7 | Task 8 | Task 9 | Task 10 |
|----------|--------|--------|--------|--------|--------|--------|--------|--------|--------|---------|
| DSL      | 0.645 ± 0.05 | 0.595 ± 0.06 | 0.681 ± 0.05 | 0.714 ± 0.05 | 0.687 ± 0.06 | 0.695 ± 0.06 | 0.733 ± 0.06 | 0.766 ± 0.07 | 0.762 ± 0.06 | 0.779 ± 0.06 |
| ISL      | 0.645 ± 0.05 | 0.596 ± 0.06 | 0.681 ± 0.05 | 0.714 ± 0.05 | 0.687 ± 0.06 | 0.697 ± 0.06 | 0.734 ± 0.06 | 0.767 ± 0.07 | 0.762 ± 0.06 | 0.783 ± 0.06 |
| OCSL     | 0.641 ± 0.05 | 0.592 ± 0.05 | 0.707 ± 0.04 | 0.749 ± 0.04 | 0.686 ± 0.06 | 0.692 ± 0.06 | 0.736 ± 0.05 | 0.765 ± 0.07 | 0.759 ± 0.06 | 0.780 ± 0.06 |
| JOCSL    | 0.634 ± 0.05 | 0.612 ± 0.05 | 0.717 ± 0.05 | 0.755 ± 0.05 | 0.696 ± 0.06 | 0.713 ± 0.06 | 0.750 ± 0.06 | 0.783 ± 0.07 | 0.800 ± 0.07 | 0.801 ± 0.05 |

Fig. 7. Classification Accuracy of 10 tasks on the LMMD Dataset. Each figure shows performance of one task; the y-axis represents classification accuracy, and the x-axis represents index of grid-searched configurations. All results are averaged over 20 random choices of training examples.

Fig. 8. F1 Score of 10 tasks on the LMMD Dataset. Each figure shows performance of one task; the y-axis represents f1 score, and the x-axis represents index of grid-searched configurations. All results are averaged over 20 random choices of training examples.
prior studies [10], [11], [12], where binding profiles are indirectly used as features to train models for predicting unknown profiles e.g., profiles are used to construct a kernel matrix based on which standard supervised learning is performed. The presented study assumes training samples have positive and unknown features, and do not consider and address the missing pattern in testing data. Whether and how addressing the latter problem could further improve performance remains an open question. We do not study hyper-parameter optimization for JOCSL neither, and one approach may be adding these parameters in the learning process as in [13], [14].

All examined approaches in this paper use ridge regression as the base prediction model, for fair comparison and due to the popularity of this model. By comparison, our key observation is that approach based on the u.p.n. assumption performs worse than those not based on the assumption; this is an indirect evidence that the u.p.n. assumption is ineffective. Our results suggest one may improve prediction performance by applying o.c.c.f. methods to recover unreported profiles; the results also suggest jointly recovering profiles and learning predictive models may further improve performance.

It should be mentioned the presented study is partly inspired by our earlier study on social network analysis [15]. When clustering nodes on social graphs, we notice a sparse social graph may be interpreted as incomplete but one usually has no idea which edges are missing. If we model the graph by a binary adjacent matrix of its nodes, with value 1 indicating an edge between two nodes and 0 otherwise, then the adjacent matrix is the incomplete feature matrix considered in this paper (with the same missing pattern). In [16], we are investigating the impact of such missing pattern on the clustering task, and start to wonder how this pattern may affect the supervised learning task; this motivates the presented study.

| Approach | Task |
|----------|------|
| DSL      | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    | 10   |
|          | 0.327 ± 0.04 | 0.328 ± 0.06 | 0.371 ± 0.04 | 0.396 ± 0.04 | 0.389 ± 0.05 | 0.405 ± 0.04 | 0.397 ± 0.03 | 0.429 ± 0.05 | 0.405 ± 0.04 | 0.415 ± 0.05 |
| ISL      | 0.327 ± 0.04 | 0.328 ± 0.06 | 0.371 ± 0.04 | 0.396 ± 0.04 | 0.389 ± 0.05 | 0.405 ± 0.04 | 0.397 ± 0.03 | 0.429 ± 0.05 | 0.406 ± 0.05 | 0.419 ± 0.04 |
| OCSSL    | 0.316 ± 0.05 | 0.318 ± 0.06 | 0.402 ± 0.05 | 0.438 ± 0.05 | 0.389 ± 0.05 | 0.405 ± 0.04 | 0.403 ± 0.03 | 0.429 ± 0.03 | 0.403 ± 0.04 | 0.417 ± 0.04 |
| JOCSL    | 0.331 ± 0.04 | 0.377 ± 0.05 | 0.399 ± 0.05 | 0.424 ± 0.05 | 0.384 ± 0.05 | 0.414 ± 0.04 | 0.397 ± 0.05 | 0.430 ± 0.05 | 0.403 ± 0.04 | 0.414 ± 0.05 |

6 Conclusion

In this paper, we present empirical evidence that the popular unreported-profile-is-negative (u.p.n.) assumption can be ineffective in predictive cheminformatics. We show supervised learning approach based on this assumption consistently performs worse than approaches that are not based on the assumption and try to recover unreported profiles. Technically, our problem is a new missing feature problem, which we called learning with positive and unknown features. We propose a novel approach that jointly recovers missing features and learns predictive model; our experimental results show this approach outperforms its counterpart that separately recovers missing features and learns predictive model.

APPENDIX A

Review: One-Class Collaborative Filtering

The one-class collaborative filtering (o.c.c.f.) problem was arguably first coined by Pan [2]. It aimed to predict users’ binary feedbacks on products (e.g., ‘like’ or ‘dislike’) based on their known feedbacks on other products, but faced the challenge that only positive feedbacks were known—for instance, the system only showed which users ‘like’ which products, but not who ‘dislike’ which products. Unlike traditional solutions that confidently assumed all unknown feedbacks were negative, Pan proposed to weight this confidence and incorporated the weight into the matrix-factorization based prediction method. In the same year, a similar problem was considered by Hu [17] under slightly different context. It seemed both Pan’s and Hu’s works were considered to pioneer the o.c.c.f. problem in the literature.

A main line of research cast o.c.c.f. as a matrix completion problem (i.e., completing the missing feedbacks in user-product feedback matrix), and adopted the matrix factorization technique for completion. The technique factorizes an incomplete feedback matrix into a user-specific factor and a product-specific factor under proper constraints, and treated the product of these factors as the completed feedback matrix. Representative works include both Pan and Hu’s approaches, where unknown feedbacks were treated as negative but with weighted confidence; Pan [18] also proposed a low-rank approximation method to improve the scalability of the matrix factorization technique; Sindhwani [19] proposed to jointly estimate the factors and predict the unknown feedbacks based on the non-negative matrix factorization technique; Vinagre [9] proposed to sample for each positive feedback a set of unknown feedbacks and treat them as negative; Zhao [20] replaced the user-specific factors with one projection matrix per user, which embodies the user’s personal item preference; Liu [21] introduced a boosting framework which builds each factorization model based on a permuted feedback matrix.

For enhancing the performance of matrix factorization techniques, several researchers proposed to incorporate side information, such as users’ search query history, purchasing and browsing activities as proposed by Li [22], the user and item profiles as proposed by Fang [23], and the user-user and item-item graphs as proposed by Yao [24].

Another line of research cast o.c.c.f. as a problem of ranking products under proper constraints induced from known feedbacks, and adopted the Bayesian method for ranking. The idea was first proposed by Rendle [25], who made the pairwise assumption that each user’s unobserved products should rank no higher than his observed products; Pan [26]
extended this assumption from a single user to a group of users for injecting richer user interactions, and Xu [27] replaced this assumption with one that biases ranking by user-generated content; Yao [28] proposed to incorporate meta-data on products to improve the ranking performance on sparse feedback matrix. The scalability of this set of approaches was addressed by Paquet [29] based on a distributed learning framework.

In addition to the above two lines of research, other feedback predictions approached included nearest neighbor as proposed by Pappas [30] and Verstrepen [31], and LRece proposed by Sedhain [7]. Recently, a theoretical study was presented by Hsien [32], where he proved upper error bounds for recovering such incomplete feedback matrix.

It is also noted o.c.e.f has found broad applications in recommending scientific articles [33], Top-N product recommendation [34], predicting drug-target interaction [35], location recommendation [36] and job recommendation [37].

While the literature of o.c.e.f continues to grow rich, we did not notice its interplay with feature-based classification problems, although many incomplete feature matrices may inherit the one-class nature. It would be helpful to understand if o.c.e.f would suffice to address these problems, and what improvements could be made. Our study in this paper is an attempt to advance the research in this direction.

APPENDIX B

REVIEW: POSITIVE AND UNLABELED LEARNING

The research on positive and unlabeled learning (p.u.l.) problem has a long history. The problem considers learning to classify data into a positive class and a negative class, but faces the challenge that training sample only contains positive data and (usually) unlabeled data.

We traced the p.u.l. research back to Muggleton’s work in 1997 [38], where he proved the error bound of learning from positive data alone. Under the p.u.l. setting, Denis proved DNF and decision tree were PAC learnable [39] and had the respective algorithms developed in [40], [41].

A notable line of p.u.l. research focused on developing two-step solutions – in the first step, an identifier selected a subset of ‘reliable’ unlabeled data and confidently assumed they were negative data; in the second step, a classifier was trained (iteratively) with both positive data and the identified negative data. For instance, Liu [42] used naive Bayes model as both the identifier and the classifier, and proved the sample complexity of the developed learner; Yu [43] used 1-DNF as the identifier and SVM as the classifier, and speeded up his approach in [44] and addressed its over-iteration problem in [45]; Li [46] used Ricchio method as the identifier for its robustness and SVM as the classifier; while previous studies largely ignored unreliable negative data, Xiao [47] added them into learning by carefully weighting these data based on their similarities.

A more recent line of research assumed all unlabeled data were negative but with weighted confidence. For instance, Lee [48] proposed a weighted logistic regression that treated all unlabeled data as negative data with label noise; Liu [49] proposed a SVM-style learner which treats all unlabeled data as negative but suffers weighted loss (the learner was showed equivalent to a two-step method with naive bayes as the identifier and SVM as the classifier); a generative version of both above approaches was proposed by Zhang [50], who also improved the SVM-style learner in [51]; Elkan [52] proposed a principled scheme to weight the unlabeled data under proper assumptions.

A group of studies focused on addressing the challenge that positive data could be scarce. In the context of text classification, Denis [53] addressed this problem by first mapping the documents to a layer of words and then to the class labels, provided that the positive class prior was known. He later extended this method to a multi-view version in [54], and Calvo [55] also refined this method on its classifier and prior probability estimation. Liu [56] proposed a framework similar to [53], but instead carefully selected those key words that were manually associated with class labels; he later relaxed the manual labeling in this method to automatic labeling in [57]; Fung [58] tackled the scarce positive data challenge by additionally labeling reliable positive data from the unlabeled data set.

We noted p.u.l. had been cast and addressed from an outlier detection point of view. A popular solution was the one-class SVM proposed by Scholkopf [59], which finds a decision boundary that separates the positive data from the origin; Manevitz [60] refined this idea by replacing the origin with a neighborhood of the origin that consists of unlabeled ones, and Vert [61] refined its error estimation by generating negative data [63]; this idea was also adopted by Manevitz [64] who developed a one-class neural network which restricts the search space for covering positive data.

The p.u.l. problem has also been studied jointly with other machine learning topics. For instance, He [65] studied p.u.l. with uncertain data; Pelckmans studied p.u.l. in a transductive setting [66]; Li studied p.u.l. in the online setting [67], [68]; Ghasemi proposed active solution for p.u.l. [69]; Mordelet proposed ensemble learning solution for p.u.l. [70]; Li [71] addressed the problem that distributions of positive data may be different in the observed training sample and the unlabeled data set; more recently, Plessis [72] suggested that in p.u.l. non-convex loss function is better than convex loss function since the latter may mislead the classification boundary; (we were all relieved) he later corrected this argument by finding a convex formula which does not mislead the boundary [73].

Finally, there has been an increasing application of p.u.l., ranging from image retrieval [74], remote sensing [75], blog classification [76], graph classification [77], deceptive review detection [78], [79], [80], name entity recognition [81], entity-attribute query [5], cross-modal retrieval [82], activity recognition [83], computer security event extraction [84] to bioinformatics [6], [85], [86].

While extensive research has been done on the p.u.l., to our knowledge they all focused on positive and unknown labels, but no work considered positive and unknown features (which is the focus on this paper). Under different settings, these once ‘labels’ could be used as ‘features’ for other
tasks, and it would be helpful to understand their impact on learning. Our study in this paper is an attempt to advance the research in this direction.

B.1 P.U.L. in Cheminformatics

Studies of PUL learning in cheminformatics can be broadly categorized into (i) drug-drug, protein-protein or drug-target interactions [87] [88] (ii) disease-gene identification [86] [89].

In the first category, drug-drug interactions (DDI) or drug-target (DTI) studies are vital when more than one drug is administered together or each drug has the potential to modulate more than one target. Known interactions are utilized in a PUL setting to infer new associations since verified negative interactions are very few. Hameed et al., proposed a PUL method to identify P450 cytochrome dependent and P450 cytochrome independent DDIs by integrating GSOM (growing self-organizing maps) and SVM [90]. The GSOM is used to cluster drugs to identify potential negative DDIs and then use an SVM classifier to build a model using positive DDIs and the identified negative DDIs. Cheng et al., used biased-SVM to predict compound-protein pairs in which the strategy is to use a pair of spherical hyperplanes such that the inner sphere tries to accommodate as many positive interactions as possible and the outer plane tries to push the negative samples out of the sphere [91]. Tsai et al., adopted a similar data level preprocessing to identify likely positive samples from U and use SVM to identify protein-protein interactions [92]. The second category of work that utilizes PUL is to identify disease genes. Yang et al., [93] proposed an ensemble PU method that first identifies reliable negative and positive samples from U and estimates the weight of the rest of the samples in U by a random walk approach. They exploit multiple biological spaces (gene expression, GO similarity and PPI) by building a network on each one of them to estimate the gene weights. The gene weights obtained from each network is then used to learn a classifier using positive and weighted unlabeled samples. They use weighted KNN, weighted Naive Bayes and SVM and combine the outputs from these classifiers for final predictions. ProDiGe is a state of the art algorithm that ranks genes for disease in a multi-task PUL setting. This model enforced similar diseases to share gene information not just across a single source but across multiple sources where each source characterizes a gene [94]. The gene information was shared in a weighted manner across diseases where the weights are proportional to the similarity of diseases.

REFERENCES

[1] J. Alvarsson, M. Eklund, O. Engkvist, O. Spjuth, L. Carlson, J. E. Wikberg, and T. Noeske, “Ligand-based target prediction with signature fingerprints,” J. Chemical Inf. Model., vol. 54, no. 10, pp. 2647–2653, 2014.
[2] R. Pan, Y. Zhou, B. Cao, N. N. Liu, R. Lukose, M. Scholz, and Q. Yang, “One-class collaborative filtering,” in Proc. Int. Conf. Data Mining, 2008, pp. 502–511.
[3] F. Cheng, C. Liu, J. Jiang, W. Lu, W. Li, G. Liu, W. Zhou, J. Huang, and Y. Tang, “Prediction of drug-target interactions and drug repositioning via network-based inference,” PLoS Comput. Biol., vol. 8, no. 5, 2012, Art. no. e1002303.
[4] R. J. Little and D. B. Rubin, Statistical Analysis with Missing Data. Hoboken, NJ, USA: Wiley, 2014.
[5] A. Haleyv, N. Noy, S. Sarawagi, S. E. Whang, and X. Yu, “Discovering structure in the universe of attribute names,” in Proc. 25th Int. Conf. World Wide Web Conf, 2016, pp. 939–949.
[6] P. Yang, S. J. Humphrey, D. E. James, Y. H. Yang, and R. Jothi, “Positive-unlabeled ensemble learning for kinase substrate prediction from dynamic phosphoproteomics data,” Bioinf., vol. 32, no. 2, pp. 259–262, 2016.
[7] S. Sedhain, A. K. Menon, S. Sanner, and D. Braziana, “On the effectiveness of linear models for one-class collaborative filtering,” in Proc. 13th AAAI Conf. Artif. Intell., 2016, pp. 229–235.
[8] Y. Koren, R. Bell, and C. Volinsky, “Matrix factorization techniques for recommender systems,” Comput., vol. 42, no. 8, pp. 33–37, 2009.
[9] J. Vinagre, A. M. Jorge, and J. Gama, “Collaborative filtering with recommender-based negative feedback,” in Proc. 30th Annu. ACM Symp. Appl. Comput., 2015, pp. 963–965.
[10] T. van Laarhoven, S. B. Nabuurs, and E. Marchiori, “Gaussian interaction profile kernels for predicting drug-target interaction,” Bioinf., vol. 27, no. 21, pp. 3036–3043, 2011.
[11] T. van Laarhoven and E. Marchiori, “Predicting drug-target interactions for new drug compounds using a weighted nearest neighbor profile,” Proc. Os Ouest, 2012.
[12] J.-P. Mei, C.-K. Kwoh, P. Yang, X.-L. Li, and J. Zheng, “Drug–target interaction prediction by learning from local information and neighbors,” Bioinf., vol. 29, no. 2, pp. 238–245, 2013.
[13] Y. Zhang, B. Cao, and D. Y. Yeung, “Multi-domain collaborative filtering,” in Proc. 26th Conf. Uncertainty Artif. Intell., 2010, pp. 725–732.
[14] A. Klimi, G. Bouchard, A. Tripathi, et al., “Group-sparsme embeddings in collective matrix factorization,” in Proc. Int. Conf. Learn. Representations, 2014.
[15] Y. Yang, C. Lan, X. Li, B. Luo, and J. Huan, “Automatic social circle detection using multi-view clustering,” in Proc. Conf. Inf. Knowl. Manage., 2014, pp. 1019–1028.
[16] C. Lan, Y. Yang, X. Li, B. Luo, and J. Huan, “Learning social circles in egocentric networks based on multi-view network structure,” IEEE Trans. Knowl. Data Eng., vol. 29, no. 8, pp. 1681–1694, Aug, 2017.
[17] Y. Hu, Y. Koren, and C. Volinsky, “Collaborative filtering for implicit feedback datasets,” in Proc. Int. Conf. Data Mining, 2008, pp. 263–272.
[18] R. Pan and M. Scholz, “Mind the gaps: Weighting the unknown in large-scale one-class collaborative filtering,” in Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2009, pp. 667–676.
[19] V. Sindhwani, S. S. Bucak, J. Liu, and A. Josifovski, “One-class matrix completion with low-density factorizations,” in Proc. Int. Conf. Data Mining, 2010, pp. 1055–1060.
[20] T. Zhao, J. McAuley, and I. King, “Improving latent factor models via personalized feature projection for one class recommendation,” in Proc. 24th ACM Int. Conf. Knowl. Manage., 2015, pp. 821–830.
[21] Y. Liu, F. Zhao, A. Sun, and C. Mao, “A boosting algorithm for item-based recommendations with implicit feedback,” in Proc. 24th Int. Conf. Artif. Intell., 2015, pp. 1792–1798.
[22] Y. Li, J. Hu, C. Zhai, and Y. Chen, “Improving one-class collaborative filtering by incorporating rich user information,” in Proc. 19th ACM Int. Conf. Knowl. Manage., 2010, pp. 959–968.
[23] Y. Fang and L. Si, “Matrix co-factorization for recommendation with rich side information and implicit feedback,” in Proc. 2nd Int. Workshop Info. Heterogeneity Fusion Recommender Syst., 2011, pp. 65–69.
[24] Y. Yao, H. Tong, G. Yan, F. Xu, X. Zhang, B. K. Szymanski, and J. Lu, “Dual-regularized one-class collaborative filtering,” in Proc. 23rd ACM Int. Conf. Knowl. Manage., 2014, pp. 759–768.
[25] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, “Bpr: Bayesian personalized ranking from implicit feedback,” in Proc. 25th Conf. Uncertainty Artif. Intell. Conf. Uncertainty Artif. Intell., 2009, pp. 452–461.
[26] W. Pan and L. Chen, “Gbpr: Group preference based bayesian personalized ranking for one-class collaborative filtering,” in Proc. 23rd Int. Joint Conf. Artif. Intell., pp. 2691–2697.
[27] S. Xu, S. Wu, and L. Wang, “Personalized semantic ranking for collaborative recommendation,” in Proc. 38th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2015, pp. 971–974.
[28] J. Yao, H. He, W. Yang, Y. Zhang, and J. Cao, “Collaborative topic ranking: Leveraging item meta-data for sparsity reduction,” in Proc. 29th AAAI Conf. Artif. Intell., 2015, pp. 374–380.
[29] U. Paquet and N. Koenigstein, “One-class collaborative filtering with random graphs,” in Proc. 22nd Int. Conf. World Wide Web Conf., 2013, pp. 999–1008.
[80] Y. Ren, D. Ji, and H. Zhang, “Positive unlabeled learning for deceptive reviews detection,” in Proc. Conf. Empirical Methods Natural Lang. Process., 2014, pp. 488–498.

[81] R. Al-Rfou, V. Kulkarni, B. Perozzi, and S. Skiena, “Polyglot-ner: Massive multilingual named entity recognition,” in Proc. SIAM Conf. Data Mining, 2015, pp. 386–394.

[82] A. K. Menon, D. Surian, and S. Chawla, “Cross-modal retrieval: A pairwise classification approach,” in Proc. SIAM Conf. Data Mining, 2015, pp. 199–207.

[83] L. T. Nguyen, M. Zeng, P. Tague, and J. Zhang, “I did not smoke 100 cigarettes today!: Avoiding false positives in real-world activity recognition,” in Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput., 2015, pp. 1053–1063. [Online]. Available: http://doi.acm.org/10.1145/2750858.2804256

[84] A. Ritter, E. Wright, W. Casey, and T. Mitchell, “Weakly supervised extraction of computer security events from twitter,” in Proc. 24th Int. Conf. World Wide Web, 2015, pp. 896–905.

[85] C. Wang, C. Ding, R. F. Meraz, and S. R. Holbrook, “Psol: A positive sample only learning algorithm for finding non-coding rna genes,” Bioinf., vol. 22, no. 21, pp. 2590–2596, 2006.

[86] P. Yang, X.-L. Li, J.-P. Mei, C.-K. Kwoh, and S.-K. Ng, “Positive-unlabeled learning for disease gene identification,” Bioinf., vol. 28, no. 20, pp. 2640–2647, 2012.

[87] Z. Cheng, S. Zhou, and J. Guan, “Computationally predicting protein-ma interactions using only positive and unlabeled examples,” J. Bioinf. Comput. Biol., vol. 13, no. 3, 2015, Art. no. 1541005.

[88] P. Yang, S. J. Humphrey, D. E. James, Y. H. Yang, and R. Jothi, “Positive-unlabeled ensemble learning for kinase substrate prediction from dynamic phosphoproteomics data,” Bioinf., vol. 32, pp. 252–259, 2015.

[89] X.-M. Zhao, Y. Wang, L. Chen, and K. Aihara, “Gene function prediction using labeled and unlabeled data,” BMC Bioinf., vol. 9, no. 1, 2008, Art. no. 57.

[90] P. N. Hameed, K. Verspoor, S. Kusljic, and S. Halgamuge, “Positive-unlabeled learning for inferring drug interactions based on heterogeneous attributes,” BMC Bioinf., vol. 18, no. 1, 2017, Art. no. 140.

[91] Z. Cheng, S. Zhou, Y. Wang, H. Liu, J. Guan, and Y.-P. P. Chen, “Effectively identifying compound-protein interactions by learning from positive and unlabeled examples,” IEEE/ACM Trans. Comput. Biol. Bioinf., vol. 15, no. 6, pp. 1832–1843, Nov./Dec. 2016.

[92] R. T.-H. Tsai, H.-C. Hung, H.-J. Dai, Y.-W. Lin, and W.-L. Hsu, “Exploiting likely-positive and unlabeled data to improve the identification of protein-protein interaction articles,” BMC Bioinf., vol. 9, no. 1, 2008, Art. no. S3.

[93] P. Yang, X. Li, H.-N. Chua, C.-K. Kwoh, and S.-K. Ng, “Ensemble positive unlabeled learning for disease gene identification,” PloS One, vol. 9, no. 5, 2014, Art. no. e97079.

[94] F. Mordelet and J.-P. Vert, “Prodige: Prioritization of disease genes with multitask machine learning from positive and unlabeled examples,” BMC Bioinf., vol. 12, no. 1, 2011, Art. no. 1.

Chao Lan received the bachelor’s degree from the Nanjing University of Posts and Telecommunications (NUPT), in 2008, the master’s degree in pattern recognition and intelligent system from NUPT, China, in 2011, advised by Xiaoyuan Jing, and the PD degree in computer science from the University of Kansas, in 2017, advised by Jun (Luke) Huan. He is an assistant professor with the Department of Computer Science, University of Wyoming. He works on machine learning and data mining.

Sai Nivedita Chandrasekaran received the bachelor’s degree from Anna University, India, in 2006, the master’s degree in electrical engineering from New Jersey Institute of Technology, in 2007, and the PhD degree in computer science from the University of Kansas, in 2017, advised by Jun (Luke) Huan. She works on machine learning and its applications in cheminformatics. Her particular research interests include multi-task learning and multi-view learning.

Jun Huan received the PhD degree in computer science from the University of North Carolina. He is a professor with the Department of Electrical Engineering and Computer Science, University of Kansas. He directs the Data Science and Computational Life Sciences Laboratory, KU Information and Telecommunication Technology Center (ITTC). Dr. Huan works on data science, machine learning, data mining, big data, and interdisciplinary topics including bioinformatics. He has published more than 120 peer-reviewed papers in leading conferences and journals and has graduated more than 10 graduate students including seven PhDs.

For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/csdl.