Drug Target Interaction Prediction with Non-random Missing Labels

Sheng Ni, Chen Lin, Xiangxiang Zeng
Department of Computer Science, Xiamen University, Xiamen, China
Yun Liang
Department of Information ,South China Agricultural University ,Guangzhou, China

Abstract—**Drug-Target Interaction (DTI) prediction is a very important direction in bioinformatics and can be used for the development of new drugs. At the same time, for existing drugs, the use of DTI prediction methods can find new targets, greatly shortening the development cycle of new drugs, and has become an important method for drug development. In our research, we assume the unknown DTIs are labels that are missing not at random. This assumption is different from the previous papers in which the unknown label samples is treated as a negative sample. For example, negative DTI labels are more likely to be missing because biomedical researchers prioritize to study DTIs that are more likely to be positive. We introduce a novel probabilistic model, Factorization with Non-random Missing Labels (FNML), for DTI prediction. FNML models the generative process for the DTI labels (i.e. the labels are positive or negative) and responses (i.e. the labels are observed or missing). In particular, the probability of observing or missing a label is associated with the sign of the label. In order to further reduce prediction variance and improve prediction accuracy on highly imbalanced DTI dataset, we present FNML-EN, an ensemble scheme which is designed specifically for FNML model. Experimental results on the latest DTI database show that FNML-EN outperforms state-of-the-art methods. We also conduct comprehensive experiments to validate the robust performance of the proposed models.**

Index Terms—**Missing Not At Random, Drug Target Interaction Prediction, Probabilistic Factor Models**

I. Introduction

FROM discovery to promotion to the market, a drug usually costs hundreds of millions of dollars a few years. Target identification and verification is the first step in the new drug development process. How to quickly and effectively identify a drug target has become a research hotspot in academia and industry.

As biochemical experimental methods for DTI identification are extremely costly and time-consuming, computational DTI prediction methods have received a growing popularity in literature. Traditional computational methods to predict DTIs mainly include ligand-based methods [1] and molecule docking methods [2]. Ligand-based methods are ineffective when target proteins have little binding ligands , while molecular docking methods are computationally costly and fail to offer accurate predictions when 3D structures of target proteins are not available [3]. To overcome these problems, many machine learning-based methods have been proposed for inferring DTI. The majority of existing machine learning-based methods treat DTI prediction as a binary classification task, where known DTIs are labeled as positive and unknown DTIs are labeled as negative [4]. The researchers extracted features from drug and target, then training models such as SVM, Logistic Regression, etc. to establish a classifier.

However, due to the particularity of drug target data, it is difficult for traditional classifiers to obtain a better effect. First, there are a large number of unknown DTIs in the drug target data. Most of the papers consider the samples of these unknown DTIs to be negative samples. Obviously, this is unreasonable because there are a large number of positive samples among the samples of these unknown DTIs. Second, there is very little positive sample data in the data. For example, In the data used in this paper, the proportion of positive samples is less than 1%. In such highly unbalanced data, most of the classifiers will fail. To address the imbalanced problem arised from the binary classification scheme, many research has attempted to extract a subset of reliable negative samples, e.g. by random sampling [5] or by **P**ositive **U**nlabel **L**earning (PU Learning) [6].

Instead of labeling the unknown DTIs as negative, we argue that it is more natural to consider the unknown DTIs, i.e. DTIs that are neither identified in vivo to be positive nor experimentally validated to be negative (non-interacting drug-target pairs), as missing labels. Furthermore, our assumption in this work is that labels are not missing at random. This is an intuitive and reasonable assumption, because researchers will use their domain expertise to filter DTIs with a high possibility to be positive and prioritize validations for these DTIs in vivo. For example, researchers find the efficacy target of a drug based on principles of biochemistry, biophysics, genetics and chemical biology. If ample evidences exist to support positive interactions with the target, then the possibility of a positive DTI is high, and the researchers are likely to conduct in vivo experiments. On the contrary, drug target interactions that are less likely to be positive are more likely to be ignored by researchers and their labels are likely to be missing.

A. Related Work

Many traditional machine learning methods simply build the classifier through the chemical structure of the drug and the sequence of the protein. As people's research on drugs and diseases continues to deepen, other information can be combined in the process of predicting drug targets, such

e-mail: chenlin@xmu.edu.cn

as drugs and side effects, drugs and diseases, targets and diseases. For instance, DTINet [5] combines drugs, diseases, side effects and other information to learn low-dimensional feature representations of drug and targets and then applies inductive matrix completion. HNM [7] integrates information through drug, target and disease to construct a three-layer heterogeneous network. After that, the strength of each drug-target pair is calculated by an iterative algorithm.

One component of our work (i.e labels are generated by feature vectors learnt and fused from heterogenous information networks) is inspired by a recent work DTINet [5]. However, there are three key differences between our work and DTINet. (1) DTINet is based on deterministic matrix factorization, our work is based on probabilistic factor models. For example, the hidden feature space mapping matrix, labels, and responses are all random variables. This setting enables the FNML model to regulate the parameters (i.e. hidden feature space mapping matrix) by introducing appropriate priors. Therefore, performance on sparse dataset is improved. (2) DTINet is based on randomly missing responses, i.e. it samples uniformly a set of unknown DTIs as negative sample, while FNML is based on missing not at random theories. Statistical theory in [9] shows that applying a model based on missing at random assumptions can lead to biased parameter estimation on data sets with missing not at random entries. (3) DTINet adopts only a subset of unknown DTIs to preserve a balanced number of positive and negative samples, while our model uses all information in the data set.

We also want to distinguish our work with another line of research. Usually only positive DTIs are deposited in known databases. Due to the lack of negative samples, PU learning has been employed in DTI identification, e.g. to facilitate negative sample extraction [6]. PU learning does not explicitly associate the status of an instance (i.e. being labeled or unlabeled) with the value of its label. We also want to mention here that, although we experiment with datasets where only positive DTIs are deposited, FNML is extendable without difficulty to databases where positive and negative DTIs are available. Thus our model is applicable in more scenarios.

B. Contribution

Our first contribution in this work is a novel **F**actorization with **N**on-random **M**issing **L**abels model (FNML). To the best of our knowledge, this is the first time missing not at random theory is applied in DTI identification. The inputs of FNML are feature vectors of drugs and targets, the partially observed labels, and the fully observed responses (i.e. labels are given or missing). We allow the feature vectors to be learnt and/or integrated from heterogenous sources. The labels and responses are binary variables. The FNML model mimics the probabilistic procedures to generate labels from feature vectors and responses from labels. Specifically, the labels are related to feature vectors of both drugs and targets, and a hidden matrix mapping from the drug features to target features. The possibility of giving a response is associated with the sign of the label.

Our second contribution is FNML-EN, an ensemble learning strategy which is specifically designed for FNML. FNML-EN is efficient as it leverages the power of over-sampling in the iterative boosting framework [8]. We use both predictions of label and response by the current FNML model to sample training instances to train the next FNML model. FNML models based on different training sets are aggregated to provide the final prediction.

We conduct comprehensive experiments on the latest DTI database. Experimental results show that the FNML model outperforms state-of-the-art DTI prediction methods in terms of **A**rea **U**nder **R**eceiver **O**perating **C**haracteristic curve (AUROC) and **A**rea **U**nder **P**recision **R**ecall curve (AUPR), which are the most commonly adopted metrics to evaluate DTI prediction performance. FNML-EN further improves prediction accuracy. We also show that our models provide robust performance enhancement, despite of the input features.

II. The Proposed Method

We start with the problem definitions and notations in Sec. II-A. We then describe the proposed model FNML in Sec. II-B. Finally we present the inference algorithm in Sec. II-C.

A. Preliminaries

DTI identification is often modeled as a binary classification task. Formally, we are given $P \in \mathcal{R}^{N\times M}$ a set of DTI labels, where $p_{i,j} = 0$ indicates a negative interaction between drug i and target j, $p_{i,j} = 1$ indicates a positive DTI, the feature vectors on drug side $X \in \mathcal{R}^{N\times K}$, where $x_{i,k}$ represents drug i's weight on drug feature k, the feature vectors on target side $Y \in \mathcal{R}^{M\times L}$, where $y_{j,l}$ represents target j's weight on target feature l. The problem is to predict for a new drug-target pair $< i', j' >$, the possibility of a positive DTI $p(p_{i',j'} = 1)$.

Similar to DTINet [5], we use a compact feature expression learnt from drug and protein networks. To extracting features X, Y, we first create networks that involve drugs (for X) and proteins (for Y). We compute similarity score between each pair of nodes in the networks. Then, the diffusion component analysis (DCA) [10] is applied to learn a low-dimensional vector representation of each node of the drug network and protein network. Note here that X, Y can be extracted from a single network or an aggregation of several networks. The details of feature extraction are described in Sec. III.

In addition to the features X, Y and labels P, we make one essential modification to the problem definition. We assume that the inputs also contain responses $R \in \mathcal{R}^{N\times M}$, where $R_{i,j} = 0$ indicates an unknown DTI, $R_{i,j} = 1$ indicates a verified DTI (positive or negative). For positive responses $R_{i,j} = 1$, the labels $P_{i,j}$ are observed. For negative responses $R_{i,j} = 0$, the labels are hidden and unknown.

B. FNML Model

We use a factor model, depicted in Fig. 1. The features X, Y are in different dimensions. To associate the drug features with the target features, we introduce a hidden matrix $Z \in \mathcal{R}^{K\times L}$, where $Z_{k,l}$ is a projection that maps the drug feature k to the target feature l. We assume that Z is sampled from a Gaussian distribution,

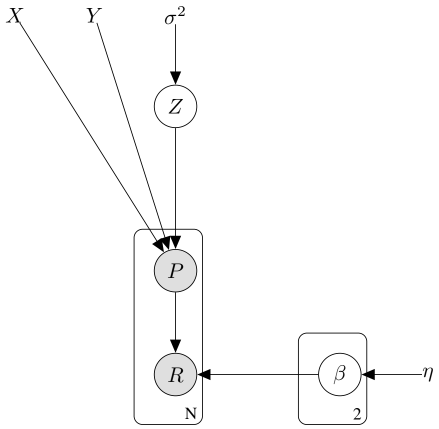


Fig. 1. Graphical Representation of the FNML model

$$\forall k, l, Z_{k,l} \sim \mathcal{N}(0, \sigma^2), \tag{1}$$

where σ^2 is the variance. We use zero mean to favor sparse feature mapping, i.e. a drug feature k is associated with a few target features.

We then assume that the binary label $P_{i,j}$ is generated from the following process:

$$\forall i, j, p(P_{i,j} = 1 | X, Y, Z) = \frac{1}{1 + \exp{(-XZY)_{i,j}}}. \tag{2}$$

The binary response is sampled from a Bernoulli distribution. The parameters of the Bernoulli distribution are related to the value of each $P_{i,j}$. Therefore we define $\beta_p \in \mathcal{R}^2, p \in \{0, 1\}$, $\forall p, \beta_{p,0} > 0, \beta_{p,1} > 0, \beta_{p,0} + \beta_{p,1} = 1$, we have:

$$\forall p \in \{0, 1\}, \beta_p \sim Beta(\eta), \tag{3}$$

$$\forall i, j, R_{i,j} \sim Bern(\beta_{P_{i,j},1}), \tag{4}$$

where $\eta \in \mathcal{R}^2$ is the hyperparameter for the Beta distribution.

C. Inference

The objective is to maximize the likelihood which consists of two terms. The first term is on partial observations, i.e. $R_{i,j} = 0$ and $P_{i,j}$ unknown. The second term is on full observations, i.e. $R_{i,j} = 1$ and known $P_{i,j}$.

$$\begin{aligned} \mathcal{L} &= \sum_{R_{i,j}=0} \log p(R_{i,j} | X, Y, \sigma^2, \eta) \\ &+ \sum_{R_{i,j}=1} \log p(R_{i,j}, P_{i,j} | X, Y, \sigma^2, \eta) \end{aligned} \tag{5}$$

Direct optimization for both terms in Equ. 5 is intractable, as they involve integration over continuous hidden variables. For example, $p(R|X, Y, \sigma^2, \eta) = \int_{P,Z,\beta} p(R|P, \beta, \eta) p(P|X, Y, Z) p(Z|\sigma^2) p(\beta|\eta)$. We employ variational inference [11] to infer the parameters. That is, we use the mean field assumption to factorize the posterior distribution:

$$q(Z, \beta, P | R, X, Y, \sigma^2, \eta) = q(P|\theta) q(Z|\mu, \upsilon) q(\beta|\rho), \tag{6}$$

It is convenient if $q(P|\theta), q(Z|\mu, \upsilon), q(\beta|\rho)$ are exponential distributions. We approximate the sigmoid function in Equ. 2 by an exponential distribution. We use the property that any sigmoid function $\sigma(\cdot)$ has a lower bound:

$$q(P|\theta) = \sigma(\theta) \geq \sigma(\zeta) \exp{((\theta - \zeta)/2 - \lambda(\zeta)(\theta^2 - \zeta^2))}, \tag{7}$$

where $\lambda(\zeta) = [\sigma(\zeta) - 1/2]/[2\zeta]$. Maximize the likelihood is equal to maximize ELBO(Evidence Lower BOund):

$$\begin{aligned} \mathcal{L}(q(Z, \beta, P)) &= E_{q(Z,\beta,P)}[\ln P(R, P, Z, \beta)] \\ &- E_{q(Z,\beta,P)}[\ln q(Z, \beta, P)] \end{aligned} \tag{8}$$

We divide the data objects into two disjoint sets, $s_1 = \{(i, j) \in \mathcal{R}^{N \times M} | R_{i,j} = 1\}$, and $s_2 = \{(i, j) \in \mathcal{R}^{N \times M} | R_{i,j} = 0\}$. First we derive the parameters of $\ln q(Z_{k,l} | \mu_{k,l}, \upsilon_{k,l})$:

$$\begin{aligned} \ln q(Z_{k,l} | \mu_{k,l}, \upsilon_{k,l}) &= \sum_{(i,j) \in s_1} E_{q(\beta)}[\ln p(R_{i,j}, P_{i,j}, Z, \beta)] \\ &+ \sum_{(i,j) \in s_2} E_{q(\beta, P_{i,j})}[\ln p(R_{i,j}, P_{i,j}, Z, \beta) \\ &+ const \end{aligned}$$

Where $const$ represents the irrelevant item. Removing irrelevant items and get:

$$\begin{aligned} \ln q(Z_{k,l} | \mu_{k,l}, \upsilon_{k,l}) &= [\sum_{(i,j) \in s_2} [(\theta_{i,j} - \frac{1}{2}) X_{i,k} * (Y^T)_{l,j}] \\ &+ \sum_{(i,j) \in s_1} \frac{1}{2} X_{i,k} * (Y^T)_{l,j}] Z_{k,l} \\ &- (\sum_{i,j} \lambda(\zeta_{ij}) * X_{i,k}^2 * {(Y^T)_{l,j}}^2 \\ &+ \frac{1}{\sigma^2}) * Z_{k,l}^2 \end{aligned}$$

Since $Z_{k,l}$ obeys a Gaussian distribution, the expectation and variance of the Gaussian distribution can be obtained:

$$\mu_{k,l} = \frac{\sum_{(i,j) \in s_2} (\theta_{i,j} - \frac{1}{2}) X_{i,k} * Y_{j,l} + \sum_{(i,j) \in s_1} \frac{1}{2} X_{i,k} * Y_{j,l}}{2 * (\sum_{i,j} \lambda(\zeta_{i,j}) X_{i,k}^2 Y_{j,l}^2 + \frac{1}{\sigma^2})}$$

$$\upsilon_{k,l} = \frac{1}{\sqrt{2 * (\sum_{i,j} \lambda(\zeta_{i,j}) X_{i,k}^2 Y_{j,l}^2 + \frac{1}{\sigma^2})}}$$

Next, we derive $\ln(\beta|\rho)$:

$$\begin{aligned} \ln q(\beta|\rho) &= \sum_{(i,j) \in s_1} E_{q(Z)} \ln p(R_{i,j}, P_{i,j}, Z, \beta) \\ &+ \sum_{(i,j) \in s_2} E_{q(Z, P_{i,j})} \ln p(R_{i,j}, P_{i,j}, Z, \beta) + const \end{aligned}$$

Expanding the two items $\sum_{(i,j)\in s_1} E_{q(Z)} \ln p(R_{i,j}, P_{i,j}, Z, \beta)$ and $\sum_{(i,j)\in s_2} E_{q(Z,P_{i,j})} \ln p(R_{i,j}, P_{i,j}, Z, \beta)$, then remove irrelevant items:

$$\begin{aligned}
\ln q(\beta|\rho) &= (\sum_{(i,j)\in s_2} \theta_{i,j} R_{i,j} + \sum_{(i,j)\in s_1} P_{i,j} R_{i,j} \\
&+ \eta_{10} - 1) \ln \beta_1 + [\sum_{(i,j)\in s_2} \theta_{i,j}(1 - R_{i,j}) \\
&+ \sum_{(i,j)\in s_1} P_{i,j}(1 - R_{i,j}) + \eta_{11} - 1] \ln(1 - \beta_1) \\
&+ [\sum_{(i,j)\in s_2} (1 - \theta_{i,j}) R_{i,j} + \sum_{(i,j)\in s_1} (1 - P_{i,j}) R_{i,j} \\
&+ \eta_{00} - 1] \ln \beta_0 + [\sum_{(i,j)\in s_2} (1 - \theta_{i,j})(1 - R_{i,j}) \\
&+ \sum_{(i,j)\in s_1} (1 - P_{i,j})(1 - R_{i,j}) \\
&+ \eta_{01} - 1] \ln(1 - \beta_0)
\end{aligned}$$

Because β obeys the Beta distribution, we can get the parameters:

$$\begin{aligned}
\rho_{0,0} &= \eta_{0,0} \\
\rho_{0,1} &= \sum_{(i,j)\in s_2} (1 - \theta_{i,j}) + \eta_{0,1} \\
\rho_{1,0} &= |s_1| + \eta_{1,0} \\
\rho_{1,1} &= \sum_{(i,j)\in s_2} \theta_{i,j} + \eta_{1,1}
\end{aligned}$$

where $|s_1|$ is the number of elements is set s_1. Next, we derive $\ln(P_{i,j}|\theta_{i,j})$:

$$\begin{aligned}
\ln q(P_{i,j}|\theta_{i,j}) &= E_{q(Z,\beta)}[\ln p(R_{i,j}, P_{i,j}, Z, \beta)] \\
&= P_{i,j} * \ln[exp(R_{i,j} * \psi(\rho_{1,0})) \\
&* exp[(1 - R_{i,j}) * \psi(\rho_{1,1})] * exp(-\psi(\rho_{1,0} \\
&+ \rho_{1,1})) * exp(X \mu_z Y)] + (1 - P_{i,j}) \\
&* \ln[exp(R_{i,j} * \psi(\rho_{0,0})) * exp[(1 - R_{i,j}) \\
&* \psi(\rho_{0,1})] * exp(-\psi(\rho_{0,0} + \rho_{0,1}))]
\end{aligned}$$

we define:

$$l_1 = exp(\psi(\rho_{1,1}) - \psi(\rho_{1,0} + \rho_{1,1}) + X_i \mu Y_j^T)$$
$$l_2 = exp(\psi(\rho_{0,1}) - \psi(\rho_{0,0} + \rho_{0,1}))$$

Then we get the estimated value of $\theta_{i,j} = \frac{l_1}{l_1 + l_2}$. Finally, for variational parameters ζ, we maximize Equ. 9:

$$\ln \sigma(\zeta_{i,j}) - \frac{\zeta_{i,j}}{2} - \lambda(\zeta_{i,j})[(X_i Z Y_j^T)^2 - (\zeta_{i,j})^2] \qquad (9)$$

Deriving the Equ. 9 and making it equal to 0. We get the update formula for the variation parameters $\zeta_{i,j} = \left| X_i \mu Y_j^T \right|$.

As shown in Alg. 1, in each iteration of the inference we alternatively optimize the variational parameters for $q(Z|\mu, \upsilon), q(\beta|\rho), q(P|\theta)$ and the parameters for the lower bound $\sigma(\zeta)$. In each iteration, we first obtain the optimal $\theta, \mu, \upsilon, \rho$ and then we update ζ. The iteration is repeated until convergence is achieved.

```
input : P, R, X, Y
output: $\mu$, $\upsilon$, $\rho$, $\theta$, $\zeta$
initialization;
repeat
    for $Z_{k,l} \in Z$ do
        $\mu_{k,l} \leftarrow \frac{\sum_{(i,j)\in s_2}(\theta_{i,j} - \frac{1}{2}) X_{i,k} * Y_{j,l} + \sum_{(i,j)\in s_1} \frac{1}{2} X_{i,k} * Y_{j,l}}{2*(\sum_{i,j} \lambda(\zeta_{i,j}) X_{i,k}^2 Y_{j,l}^2 + \frac{1}{\sigma^2})}$;
        $\upsilon_{k,l} \leftarrow \frac{1}{\sqrt{2*(\sum_{i,j} \lambda(\zeta_{i,j}) X_{i,k}^2 Y_{j,l}^2 + \frac{1}{\sigma^2})}}$;
    end
    for $\beta$ do
        $\rho_{0,0} \leftarrow \eta_{0,0}$;
        $\rho_{0,1} \leftarrow \sum_{(i,j)\in s_2} (1 - \theta_{i,j}) + \eta_{0,1}$;
        $\rho_{1,0} \leftarrow |s_1| + \eta_{1,0}$;
        $\rho_{1,1} \leftarrow \sum_{(i,j)\in s_2} \theta_{i,j} + \eta_{1,1}$;
    end
    for $(i,j) \in s_2$ do
        $l_1 = exp(\psi(\rho_{1,1}) - \psi(\rho_{1,0} + \rho_{1,1}) + X_i \mu Y_j^T)$;
        $l_2 = exp(\psi(\rho_{0,1}) - \psi(\rho_{0,0} + \rho_{0,1}))$;
        $\theta_{i,j} \leftarrow \frac{l_1}{l_1 + l_2}$;
    end
    for $(i,j) \in s_1 + s_2$ do
        $\zeta_{i,j} \leftarrow \left| X_i \mu Y_j^T \right|$;
    end
until convergence;
```

Algorithm 1: Inference for FNML

D. Model Ensemble

As previously shown, FNML model has the advantage of debasing the labels by utilizing the non-randomly missingness of responses. However, due to the severe sparsity of available responses in DTI database, the prediction of FNML can still be of high variance and over-fit the training set. To tackle this problem, we propose FNML-EN, an ensemble algorithm which is specifically designed for FNML model.

FNML-EN is inspired by the well-known boosting algorithm [8], and the SMOTE [12] oversampling technique which has been successfully applied in many imbalanced classification problems. Recall that in boosting, training instances are iteratively re-weighted on the basis of classification error. In FNML-EN, an iterative reweighing framework is also adopted. In the $t-$th round, we run FNML on the current training set to obtain the prediction of labels (i.e. $p_{i,j} = p(P_{i,j} = 1|X, Y, Z)$) and prediction of responses (i.e. $r_{i,j} = p(R_{i,j} = 1|P_{i,j}, \beta, \eta)$). We then sort each training instance (i.e. drug-target pair (i, j)) in descending order of $p_{i,j}$ to get its rank (denoted as $rank_{i,j}$). Finally, we sample with replacement positive instances based on the $s_{i,j}$, which is defined as:

$$s(i,j) = \frac{ln(\frac{rank_{i,j}}{r_{i,j}})}{\sum_{i,j} ln(\frac{rank_{i,j}}{r_{i,j}})} \qquad (10)$$

As in SMOTE [12], we will add the synthetic data point to the original set to form a new training set for the $t+1$ round. In the experiment, the number of sampled instances is equivalent to the number of positive labeled instances in the groundtruth.

For example, if there are n positive DTI pairs in the original dataset, then in each round we will have $2 \times n$ positive DTI pairs in the training set. The iteration is terminated after I rounds, and the final prediction is made by averaging the results of all FNML models.

Now let's take a closer look into the sampling weight Equ. 10. The possibility of a positive training instance being sampled is increased if it is classified wrongly (i.e. smaller $p_{i,j}$ and henceforth with larger $rank_{i,j}$). We use the predicted response to further adapt to FNML at each round. We will validate the effect of $r_{i,j}$ in our sampling strategy in Sec. III.

III. Experiment

A. Experimental Setup

Datasets. We use the same datasets as in [5]: i.e. the drug-target interaction labels are obtained from the latest version of DrugBank (version 3.0) [13]. This data set is referred to as the full data set. Only 0.18% of the drug-target interactions are labelled as positive, none is labelled as negative. As in [5], we also construct a sample dataset, where all the positive interactions are reserved and an equal number of unknown interactions are sampled to be negative. Statistics of the two data sets are shown in Tab. I.

TABLE I
Statistics of the datasets

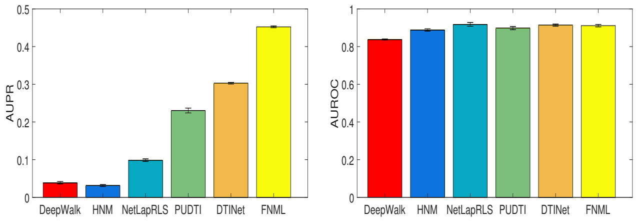
Data	#Drugs	#Targets	#Positive	#Negative	#Unknown
Full	708	1,512	1,923	0	1,068,573
Sample	708	1,512	1,923	1,923	1,066,650

We use a variety of networks to extract features X, Y. The default feature vectors X are extracted from drug structure similarity network (denoted as ds), where the similarity score between two drugs is calculated using the Tanimoto coefficient [14] according to their chemical structures; The default feature vectors Y are extracted from protein sequence similarity network (denoted as ps), which is constructed by computing the Smith-Waterman score [15] of their primary sequences. In order to evaluate model performance with different features, we also use three extra drug networks: drug-drug interaction network (dd) [13], the drug-disease network (di) [16], the drug-side-effect network (de) [17] and two protein networks: the protein-disease association network (pd) [16], the protein-protein interaction network (pp) [18].

Evaluation. Throughout the experiment section, the major evaluation metric is **A**rea **U**nder **P**recision **R**ecall curve (AUPR), which is commonly adopted in bioinformatic studies. An auxiliary evaluation metric is **A**rea **U**nder **ROC** curve (AUROC).

B. Results and Analysis

FNML Performance. We first evaluate the accuracy of DTI prediction of the proposed FNML model. The hyper-parameter settings are as follows. The number of dimensions for drug features are $K = 300$, for target features $L = 300$, hyper-parameters are $\sigma^2 = 1, \eta_0 = 1, \eta_1 = 1$. In this experiment, we use the default features X, Y. The code and data used in FNML are available at: https://github.com/517515435/FNML



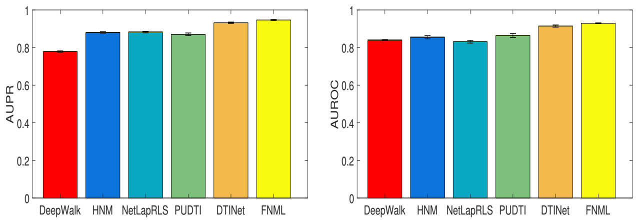
(a) AUPR (b) AUROC

Fig. 2. On the full dataset, FNML significantly boosts AUPR while obtaining comparable AUROC.

We compare our FNML model with 5 state-of-the-art methods: (1) DeepWalk [19]: a similarity-based drug-target prediction method that enhances similarity computation by deep learning method within a linked tripartite network. (2) HNM [7]: a network model in which strength between a disease-drug pair is calculated through an iterative algorithm on the heterogeneous graph that also incorporates drug-target information. (3) NetLapRLS [20]: a manifold regularization semi-supervised learning method. (4) PUDTI [6]: an SVM-based optimization model that is trained on negative samples extracted based on positive-unlabeled learning. (5) DTINet [5]: a regression model that learns feature space mapping Z by the loss function $\min_Z \sum_{i,j}(P_{i,j} - (XZY)_{i,j})^2$. We do not change the default settings for all the above comparative methods.

We perform the evaluation on two datasets. The first one is on the full dataset, i.e. we randomly segment the whole data set to 10 divisions and conduct 10-fold cross-validation. The second one is on the sample dataset, i.e. keeping the ratio of positive and negative samples to $1 : 1$, we conduct random sampling for 10 times and the reported results are averaged over the 10 sets.

The comparative performance on the full dataset is shown in Fig. 2. We can see that (1) FNML model significantly boosts the AUPR performance by 49.32%, compared with the best of state-of-the-art methods. The best comparative method is DTINet, which achieves a 30.29% AUPR. Our FNML model obtains a 45.23% AUPR. As AUPR is well regarded to be a more robust and accurate evaluation metric than AUROC [5], this observation demonstrates the potential of our model. (2) Most of the state-of-the-art methods yield very low AUPR results on the full dataset. This observation again reveals that obtaining a high AUPR performance is challenging on the full dataset. (3) In term of AUROC, the best result is obtained by NetLapRLS. However, the best comparative result is 91.78%, while FNML produces a comparable 91.12% AUROC.



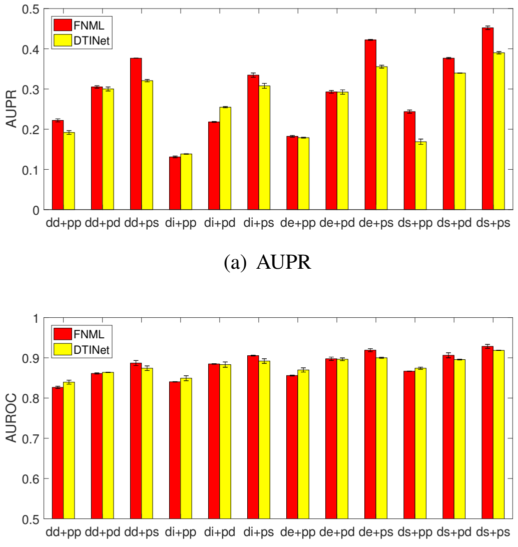
(a) AUPR (b) AUROC

Fig. 3. On the sample dataset, FNML outperforms state-of-the-art methods in terms of both AUPR and AUROC.

The comparative performance on the sample dataset is shown in Fig. 3. We can see that (1) FNML model achieves a better AUPR than all state-of-the-art methods. The best comparative method is DTINet, which achieves 93.20%. Our FNML model obtains a 94.66% AUPR. (2) Most of the state-of-the-art methods have a higher AUPR result on the sample dataset than the full dataset, due to the balanced ratio of positive and negative samples. (3) FNML model outperforms all state-of-the-art models in AUROC performance. The best comparative method is again DTINet, which achieves 91.41%. Our FNML model obtains a 92.93% AUROC. (4) Surprisingly, DeepWalk has a lowest AUPR performance on the sample set. A possible reason is that the network representation extracted by deepwalk is based on homogeneous network structure, and thus is not accurate.

FNML Performance with Different Features. We next study how FNML model performs with different features. We use various combination of X and Y as inputs. That is, we extract X from the four networks on the drug side (i.e. dd,di,de,ds) respectively, extract Y from the three networks on the protein side (i.e. pp, pd, ps) respectively, and use the 12 combinations as inputs to train the model. The predictions are tested on the full dataset.

We compare the AUPR and AUROC performance of FNML and DTINet. As shown in Fig. 4, FNML outperforms DTINet in most cases. FNML generates better AUPR results for 10 feature combinations out of 12. In term of AUROC, FNML is better for 7 feature combinations. The result shows that the performance improvement is stable. Change of feature representations does not affect FNML's ability to learn a better feature mapping space.

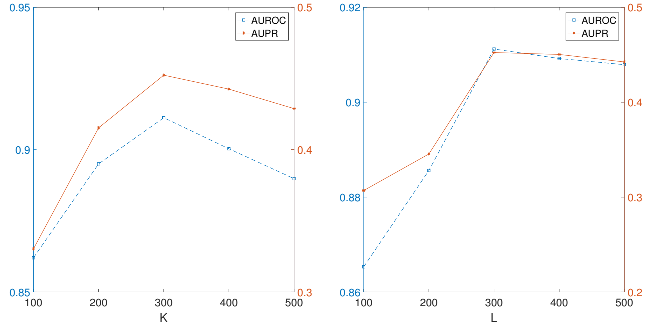


(a) AUPR

(b) AUROC

Fig. 4. FNML model consistently outperforms DTINet with different feature inputs.

Number of dimensions. We next study the effects of number of dimensions K, L. We first fix $L = 300$ and tune from $K = 100$ to $K = 500$.We can see from Fig. 5(a) that the best number of drug features is around 300. Then, we fix $K = 300$ and tune from $L = 100$ to $L = 500$. As shown in Fig. 5(b), the best number of target features is 300. An appropriate number of drug features is important. When the number of drug features is too large or too small i.e. $K \geq 400$ or $K \leq 200$, we observe a descent fall in both AUPR and AUROC. However, the model performance is less sensitive to the number of target features. For $L > 300$, AUPR and AUROC remain the same.



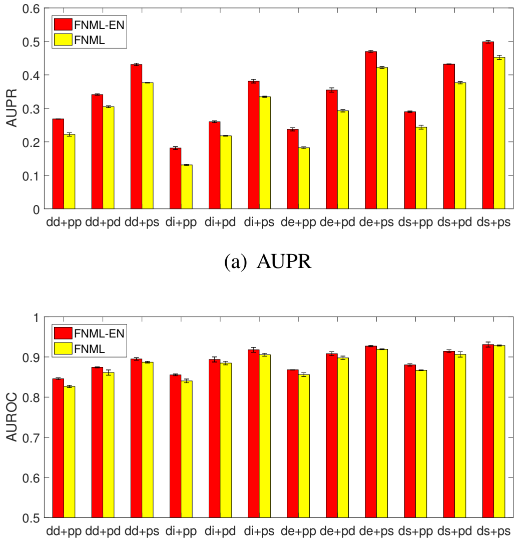
(a) K number of drug features

(b) L number of protein features

Fig. 5. AUPR and AUROC performance of our model with different number of drug and protein features.

Performance of Ensemble Model. We conduct experiment on full dataset to evaluate our proposed ensemble model. We use the same hyper-parameters settings and number of feature dimensions for drugs and proteins.

For a detailed study, we compare the AUPR and AUROC performance of FNML and FNML-EN, with different feature combinations. The results are shown in Fig. 6. We can see that our ensemble model FNML-EN generates better results in terms of AUPR and AUROC for all input features. This result shows that our ensemble model FNML-EN can robustly enhance FNML.

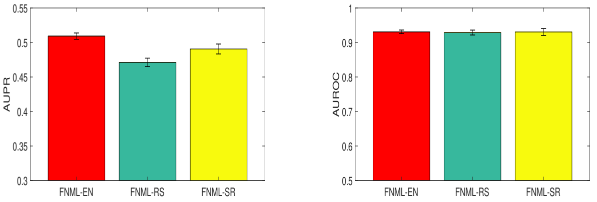


(a) AUPR

(b) AUROC

Fig. 6. FNML-EN consistently enhances FNML with different feature inputs.

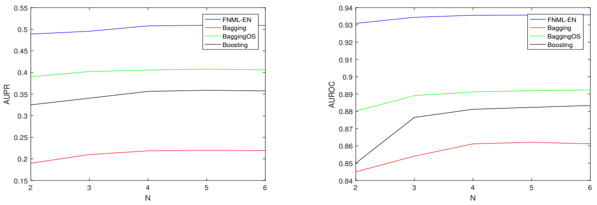
Performance of Sampling Strategy. Here we conduct an experiment to verify the effectiveness of our sampling strategy



(a) AUPR

(b) AUROC

Fig. 7. Our sampling strategy outperforms other strategys.



(a) AUPR

(b) AUROC

Fig. 8. Our ensemble model outperforms traditional ensemble models.

(Equ. 10). We compare the performances of our sampling strategy with the following two sampling strategies: (1) FNML-RS: randomly sample the same size sample from the positive sample, and then combine with the original positive sample to form a new positive sample. (2) FNML-SR: sample positive instances according to $s_1(i,j) = \frac{ln(rank_{i,j})}{\sum_{i,j} ln(rank_{i,j})}$.

The results are shown in Fig. 7. We can see that: (1) The AUROC values of the three methods are not much different. (2) Our sampling strategy outperforms FNML-RS and FNML-SR in terms of AUPR. Our FNML-EN gets a 50.92% AUPR and FNML-SR gets a 49.06% AUPR, which validate our assumption that response probability $r_{i,j}$ plays a key role in the sampling procedure.

Comparable Performance of Ensemble model. Finally, we compare our ensemble model with other ensemble models. The comparative methods are as follows. (1) Bagging [21]. We sample the training set with replacement to generate I new training sets, and use these I new training sets to train I FNML models. After that, we average the output of I FNML models to get final result. (2) Bagging With Over Simpling (BaggingOS). For a fair comparison, we also over-sample the positive instances in the training sets, and then perform bagging to output an aggregated prediction. (3) Boosting. Consider our model as a classifier and use the standard boosting framework to get the boosting ensemble of FNML model. We tune the number of weak learners (i.e. the number of iterations I in FNML-EN) from 2 to 6.

The comparative performance on full dataset is shown in Fig. 8. We can see that (1) FNML-EN achieves best AUPR and AUROC results, compared with bagging and boosting method with different numbers of weak learners I. This indicates that the proposed ensemble method is robust and superior than conventional ensemble learning methods. (2) The result of Bagging is relatively poor, because the dataset is highly imbalanced. The positive samples are not fully utilized due to the sampling procedure. (3) The result of BaggingOS is better than Bagging, because over-sampling the positive instances generates accurate predictions to fully utilize information in the training data. (4) The result of boosting is not good, indicating that treating positive and negative samples equally is not applicable to highly imbalanced data. (5) In terms of AUROC, the performance of an ensemble learner increases as the number of weak learners I increases. However, the performance of AUPR does not differ much. When $I = 4$, our model get highest AUPR (50.92%).

IV. Conclusion

We propose a novel DTI prediction model based on the assumption that unknown DTI labels are missing not at random. By associating the status of a DTI being labelled or unknown to the sign of the DTI label, our proposed FNML model can learn a better feature mapping from drug feature space to target feature space. We experimentally demonstrate that FNML outperforms state-of-the-art computational DTI identification methods. This work sheds some insights into fully exploiting the information in unknown DTIs. We further enhance the DTI prediction performance by an ensemble scheme. The ensemble scheme leverages the predictions of labels and responses by FNML in a framework which integrates over-sampling and boosting. We experimentally validate the importance of including predictions of responses in oversampling. Our future directions include improving the factorization framework and analyzing the missing mechanisms.

References

[1] Keiser, M. J. et al. Relating protein pharmacology by ligand chemistry. Nature biotechnology 25, 197206 (2007).

[2] Cheng, A. C. et al. Structure-based maximal a nity model predicts small-molecule druggability. Nat. Biotechnol. 25, 7175 (2007).

[3] Chen, X. et al. Drug-target interaction prediction: databases, web servers and computational models. Brief. Bioinform. 17, 696712 (2016).

[4] Ding, H., Takigawa, I., Mamitsuka, H., and Zhu, S. Similarity-based machine learning methods for predicting drug-target interactions: a brief review. In *Briefings in bioinformatics*, 15(5), 734-747.

[5] Luo Y, Zhao X, Zhou J, et al. A network integration approach for drug-target interaction prediction and computational drug repositioning from heterogeneous information. In *Nature Communications*, 2017, 8(1).

[6] Peng L, Zhu W, Liao B, et al. Screening drug-target interactions with positive-unlabeled learning. In *Scientific Reports*, 2017, 7(1): 8087.

[7] Wang W, Yang S, Zhang X, et al. Drug repositioning by integrating target information through a heterogeneous network model. In *Bioinformatics*, 2014, 30(20):2923-2930.

[8] Schapire R E. The Boosting Approach to Machine Learning: An Overview In *Nonlinear Estimation and Classification*. Springer New York, 2003:149-171.

[9] R. J. A. Little and D. B. Rubin. Statistical Analysis with Missing Data, 1987.

[10] Cho H, Berger B, Peng J. Diffusion Component Analysis: Unraveling Functional Topology in Biological Networks In *Research in Computational Molecular Biology*. Springer International Publishing, 2015:62-64.

[11] Bishop C M. Pattern Recognition and Machine Learning. In *Information Science and Statistics*. Springer-Verlag New York, Inc. 2006.

[12] Chawla N V, Bowyer K W, Hall L O, et al. SMOTE: synthetic minority over-sampling technique[J]. In *Journal of Artificial Intelligence Research*, 2002, 16(1):321-357.

[13] Knox C, Law V, Jewison T, et al. DrugBank 3.0: a comprehensive resource for Omics research on drugs. In *Nucleic Acids Research*, 2011, 39(Database issue):D1035.

[14] Hattori, M., Okuno, Y., Goto, S. Kanehisa, M. Development of a chemical structure comparison method for integrated analysis of chemical and genomic information in the metabolic pathways. In *Journal of the American Chemical Society* 125, 1185311865 (2003).

[15] Smith, T. F. and Waterman, M. S. Identification of common molecular subsequences. In *Journal of molecular biology* 147, 195197 (1981).

[16] Davis, A. P., Murphy, C. G., Johnson, R., Lay, J. M., Lennon-Hopkins, K., Saraceni- Richards, C., Sciaky, D., King, B. L. Rosenstein, M. C., Wiegers, T. C., et al. The comparative toxicogenomics database: update 2013. *Nucleic acids research*, 41(D1), D1104D1114.

[17] Kuhn, M., Campillos, M., Letunic, I., Jensen, L. J., and Bork, P. A side effect resource to capture phenotypic effects of drugs. *Molecular systems biology*, 6(1), 343.

[18] Keshava Prasad T S, Goel R, Kandasamy K, et al. Human Protein Reference Database–2009 update. *Nucleic Acids Research*, 2009, 37(Database issue):767-72.

[19] Zong N, Kim H, Ngo V, et al. Deep Mining Heterogeneous Networks of Biomedical Linked Data to Predict Novel Drug-Target Associations. In *Bioinformatics*, 2017, 33(15).

[20] Xia Z, Wu L Y, Zhou X, et al. Semi-supervised drug-protein interaction prediction from heterogeneous biological spaces. In *Bmc Systems Biology*, 2010, 4(S2):1-16.

[21] Breiman L. Bagging predictors. In *Machine Learning*, 1996, 24(2):123-140.