Supplementary Materials

1 Algorithm and Analysis

Algorithm. Algorithm 1 presents the formal description of the above steps. For the representation of an entity e in drug-drug-cell-line pair (e_i, e_j, e_k) (Line 2), the interaction fields $\{S_e^h\}_{h=0}^H$ of this entity is consisted of target fields from entity-protein associations A and radiant fields extending along interactions in the PPI network G (Line 3). The calculation of neighborhood representations is repeated H times (Line 4): in layer h for an entity e, we first calculate the contribution of each protein p ($p \in S_e^h$) to the entity and combine them together as the updated representation of entity (Line 5,6). The final representation of entity $\hat{e}$ is obtained by feeding all the representations $\{I_{S_e^h}\}_0^H$ into an aggregation function $aggre$ (Line 8). The predicted probability $\hat{y}_{i,j,k}$ is the difference between the therapy score s_p and the toxic score s_n (Line 11), which are computed by the final representations of drug i, drug j and cell line k (Line 9,10).

Algorithm 1 GraphSynergy algorithm

Input: Drug-drug-cell table Y; PPI network G; entity-protein associations A; interaction fields S_e ($e \in N_d \cup N_c$); trainable parameters: $\{e_d\}_{d \in N_d}$, $\{e_c\}_{c \in N_c}$, $\{e_p\}_{p \in N_p}$, $aggre()$, $\Gamma()$, $\Psi()$, $\sigma()$;
Output: $\mathcal{F}(i, j, k | Y, G, A, \Theta)$

1: **while** GraphSynergy not converge **do**
2: **for** $e \in (e_i, e_j, e_k), (i, j, k) \in Y$ **do**
3: $\{S_e^h\}_{h=0}^H \leftarrow$ Interaction-Field(e);
4: **for** $h = 0, 1, ..., H$ **do**
5: $I_{S_e^h} \leftarrow \sum_{p \in S_e^h} \hat{\pi}_p^e p$
6: $e \leftarrow I_{S_e^h}$
7: **end for**
8: $\hat{e} \leftarrow aggre(I_{S_e^0}, I_{S_e^1}, ..., I_{S_e^H})$
9: Calculate therapy score $s_p = \Gamma(\hat{e}_i, \hat{e}_j, \hat{e}_k)$
10: Calculate toxic score $s_n = \Psi(\hat{e}_i, \hat{e}_j)$
11: Calculate predicted probability $\hat{y}_{i,j,k} = \sigma(s_p - s_n)$
12: Update parameters Θ
13: **end for**
14: **end while**
15: **return** $\mathcal{F}$

Algorithm Analysis. Given the number of drug-drug-cell combinations n, the embedding dimension d, the depth of the interaction fields H, and the sample size of neighbors in each layer $\hat{S}$, the analysis of the model size and time complexity is as follows.

Model Size. For the embedding of proteins, drugs, and cancer cell lines, the size of the vector representations is $(|N_d| + |N_c| + |N_p|) \times d$. Besides, the aggregation function has the parameter of size $H \times d \times d + d$, and the size of parameter for the computation of toxic score is at most $2 \times d \times d$. Therefore, the total model size is $(|N_d| + |N_c| + |N_p|) \times d + (H + 2) \times d^2 + d$.

Time Complexity. For the contribution propagation component, the computation cost is $\mathcal{O}(n\hat{S}Hd)$. Then comes the aggregation layer with time complexity $\mathcal{O}(nHd^2)$. The time consumption for the computation of two scores is at most $\mathcal{O}(nd^2)$. And for the prediction step, the time complexity is $\mathcal{O}(d)$.

2 Settings of Hyper-parameter for Baselines

The settings of hyper-parameters for baselines are as follows. For GraRep model, the embedding dimension is set to 128, and the transition step $k = 1$. For two RW-based models, the walk length $t = 10$, the window size $w = 5$, and the embedding size $d = 128$, and the return $p = 0.25$, the In-out $q = 4$ for Node2Vec especially. For DeepSynergy model, we utilize the related proteins as the features for drugs and cell lines. The embedding size d is set to 32 for two GCN-based models, and we utilized logistic regression as final layer predicting cell-specific drug combinations. Specifically, for GCN model, we applied a two GCN layer with embedding loss decay as $1e - 5$, and drop out is set to 0.5 and 0.1 for DrugCombDB and Oncology-Screen, respectively. For KGNN model, the depth of receptive filed $d = 2$ and the neighbor sampling size of each depth equals 128.