OPEN ACCESS Freely available online

PLOS ONE

Prediction of Heterodimeric Protein Complexes from Weighted Protein-Protein Interaction Networks Using Novel Features and Kernel Functions

Peiying Ruan[1], Morihiro Hayashida[1]*, Osamu Maruyama[2], Tatsuya Akutsu[1]*

1 Bioinformatics Center, Institute for Chemical Research, Kyoto University, Gokasho, Uji, Kyoto, Japan, **2** Institute of Mathematics for Industry, Kyushu University, 744 Motooka, Nishi-ku, Fukuoka, Japan

Abstract

Since many proteins express their functional activity by interacting with other proteins and forming protein complexes, it is very useful to identify sets of proteins that form complexes. For that purpose, many prediction methods for protein complexes from protein-protein interactions have been developed such as MCL, MCODE, RNSC, PCP, RRW, and NWE. These methods have dealt with only complexes with size of more than three because the methods often are based on some density of subgraphs. However, heterodimeric protein complexes that consist of two distinct proteins occupy a large part according to several comprehensive databases of known complexes. In this paper, we propose several feature space mappings from protein-protein interaction data, in which each interaction is weighted based on reliability. Furthermore, we make use of prior knowledge on protein domains to develop feature space mappings, domain composition kernel and its combination kernel with our proposed features. We perform ten-fold cross-validation computational experiments. These results suggest that our proposed kernel considerably outperforms the naive Bayes-based method, which is the best existing method for predicting heterodimeric protein complexes.

Citation: Ruan P, Hayashida M, Maruyama O, Akutsu T (2013) Prediction of Heterodimeric Protein Complexes from Weighted Protein-Protein Interaction Networks Using Novel Features and Kernel Functions. PLoS ONE 8(6): e65265. doi:10.1371/journal.pone.0065265

Editor: Claudio M. Soares, Instituto de Tecnologica Química e Biológica, UNL, Portugal

Received February 12, 2013; **Accepted** April 23, 2013; **Published** June 11, 2013

Funding: This work was partially supported by Grants-in-Aid #22240009 and #24500361 from MEXT, Japan (http://www.mext.go.jp/english/). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript. No additional external funding received for this study.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: morihiro@kuicr.kyoto-u.ac.jp (MH); takutsu@kuicr.kyoto-u.ac.jp (TA)

Introduction

Protein complexes play crucial roles in a variety of biological processes, such as ribosomes for protein biosynthesis, molecular transmission and evolution of interactions between proteins. In fact, many proteins come to be functional only after they interact with their specific partners and are assembled into protein complexes. Hence, much effort has been made for predicting protein complexes from protein-protein interaction (PPI) networks [1–6] in bioinformatics. The Markov Cluster (MCL) algorithm [7] iteratively generates a matrix, called Markov matrix, in which each row (each column) corresponds to a protein and each element represents the relationship between two proteins. Then, MCL extracts clusters from the matrix. This algorithm is efficient also for large-scale networks because Markov matrices are calculated by matrix multiplication and exponentiation of its individual elements. The Molecular Complex Detection (MCODE) algorithm [8] gives a weight to each vertex by using a modified clustering coefficient, which is defined as edge density in a subset of neighboring vertices and the originating vertex. Then, it finds densely connected regions of molecular interaction networks based on the weighted vertices. The Restricted Neighborhood Search Clustering (RNSC) algorithm [9] separates the set of vertices into clusters by searching locally in a randomized fashion based on a cost function. After that, the clusters will be filtered according to the cluster size, density and functional homogeneity. The Protein Complex Prediction (PCP) algorithm [10] finds maximal cliques within PPI networks modified by using the functional similarity weight (FS-Weight) based on indirect interactions, and merges their cliques. These methods are intended for detecting dense subgraphs in a PPI network. Hence, they cannot find a protein complex with size two because the density is always 1.0 and the subgraph (i.e., an edge) itself is a clique even if two proteins that interact with each other do not form a complex. In addition, it is considered that any overlap rate of a predicted protein complex to a small known complex is more likely to be by chance than the same overlap rate to a larger known complex as pointed out in [11]. Most prediction methods have been evaluated for protein complexes with larger size than three excluding complexes with small sizes.

However, the majority of known protein complexes are heterodimeric protein complexes. CYC2008 [12], which is a comprehensive catalogue of 408 manually curated yeast protein complexes reliably supported by small-scale experiments, includes 172 (42%) heterodimeric protein complexes. Besides, MIPS protein complex catalog [13], which provides detailed information involved protein sequences on whole-genome analysis [14–16], contains 64 (29%) heterodimeric protein complexes excluding complexes obtained from high-throughput experiments. Hence, it is necessary to develop another method for predicting smaller complexes. Qi et al. proposed a method using a supervised Bayesian classifier [17] that has good performance for predicting protein complexes of middle sizes. The method still does not work

well for heterodimeric protein complexes because they used several features based on graph density and degree statistics. There are some approaches based on random walks on PPI networks. The Repeated Random Walks (RRW) method [18] repeatedly expands a focused cluster of proteins depending on the steady state probability of random walks with restarts from the cluster whose proteins are equally weighted. The Node-Weighted Expansion (NWE) method [19] is an extension of RRW. NWE restarts from the cluster whose proteins are weighted by the sum of the edge weights of the physical interactions with neighboring proteins, where the edge weights are obtained from the WI-PHI database [1]. Then, Maruyama [11] proposed an approach based on a naive Bayes classifier using heterogeneous genomic data for predicting heterodimeric protein complexes with features involved with protein-protein interaction data, gene expression data, and gene ontology annotations. This method outperforms other existing prediction methods, MCL, MCODE, RRW, and NWE, in F-measure for heterodimers [11] although these methods are not supervised.

To further improve the prediction accuracy for heterodimeric protein complexes, we propose a method using *C*-Support Vector Classification (*C*-SVC) with several features based on protein-protein interaction weights that are considered as reliability of interactions between proteins. The idea behind the design of feature space mappings is, for example, that the neighboring weights of a heterodimeric complex tend to be smaller than the weight inside of the complex. In addition to features based on weights, we propose feature space mappings based on the numbers of protein domains because those are considered to be functional and structural units in proteins. Furthermore, we propose a domain composition kernel based on the idea that two proteins having the same composition of domains as a heterodimeric protein complex would also form a heterodimer. We perform ten-fold cross validation, and calculate the average F-measures. The results suggest that our proposed kernel considerably outperforms the naive Bayes-based method, which is the best existing method.

Methods

The problem we address in this study is stated as follows: Given a network of protein-protein interactions, where interactions are weighted, determine whether or not two interacting distinct proteins form a protein complex with size exactly two. A network of protein-protein interactions can be considered as a graph, where vertices represent proteins and edges represent protein interactions. Let $G(V, E)$ be an undirected graph with a set V of vertices and a set E of edges, where the weight of each edge $(i,j)\in E$ is denoted by w_{ij} and represents reliability and strength of the interaction related with the edge. Actually, we use the WI-PHI database [1] as edge weights, which is derived from heterogeneous data sources, and was used in previous studies [11,18,19]. In this section, we propose several features for predicting heterodimeric protein complexes, a novel kernel matrix based on protein domain composition, and the combination kernel.

Feature Space Mapping Based on Interaction Weights

We propose simple feature space mappings based on weights of interactions, which are regarded to be reliabilities and strengths for protein-protein interactions as shown in Table 1. The basic idea for designing features is as follows. The reliability of the interaction in a heterodimeric complex should be high. In addition, the reliability of the interaction between a protein contained in a complex and a protein not contained in the complex should be low. These features are not only applied to *C*-SVC through linear kernels but are transformed to other kernel matrices using extended diffusion and label sequence kernels.

Table 1. Feature space mapping from two interacting proteins P_i, P_j and neighbors.

$(F1)$	w_{ij}
$(F2)$	$\max\left\{\max_{\{k\mid(i,k)\in E,k\neq j\}} w_{ik}, \max_{\{k\mid(j,k)\in E,k\neq i\}} w_{jk}\right\}$
$(F3)$	$\min\left\{\min_{\{k\mid(i,k)\in E,k\neq j\}} w_{ik}, \min_{\{k\mid(j,k)\in E,k\neq i\}} w_{jk}\right\}$
$(F4)$	$\max_{\{k\mid(i,k)\in E,(j,k)\in E\}} \min\{w_{ik},w_{jk}\}$
$(F5)$	$\max_{\{k_1,k_2\mid(i,k_1)\in E,k_1\neq j,(j,k_2)\in E,k_2\neq i\}} \lvert w_{ik_1}-w_{jk_2}\rvert$
$(F6)$	$\max\{\#$ domains of $P_i,\#$ domains of $P_j\}$
$(F7)$	$\min\{\#$ domains of $P_i,\#$ domains of $P_j\}$

doi:10.1371/journal.pone.0065265.t001

Consider two interacting proteins P_i and P_j corresponding to an input. Figure 1 shows an example of a subgraph with P_i, P_j, and their neighboring proteins P_k such that $(k,i)\in E$ or $(k,j)\in E$, where interactions between these proteins are shown as edges. One feature is the weight w_{ij} between proteins P_i and P_j, denoted by (F1), because the proteins in a heterodimeric protein complex should interact with each other and the weight w_{ij} should be large.

However, even if w_{ij} is large, the proteins could be included in a complex with size larger than two. Hence, we consider the weights of interactions with the neighboring proteins P_k. Since the neighboring weights of a heterodimeric complex tend to be smaller than the weight inside of the complex, we introduce the maximum of the neighboring weights denoted by (F2) as a feature.

In contrast, if the neighboring weights are larger than the weight w_{ij}, we can estimate that the proteins P_i and P_j would not form a complex but neighboring proteins and either P_i or P_j would form some complex. Thus, we introduce the minimum of the neighboring weights denoted by (F3).

Even if the maximum of the neighboring weights (F2) is large enough, the proteins P_i and P_j as well as P_i and P_k or P_j and P_k may form a heterodimeric complex. Consider the case that a protein P_k interacts with both of P_i and P_j. If two weights w_{ik} and w_{jk} are large, these proteins P_i, P_j and P_k are likely to form a complex. Besides, if w_{ij} is smaller than w_{ik} and w_{jk}, P_i, P_k and P_j, P_k independently can form a heterodimeric complex. For this reason, we introduce the maximum of smaller weights denoted by (F4).

In the discussion so far, we dealt only with the value of weights. However, differences between weights are also important for

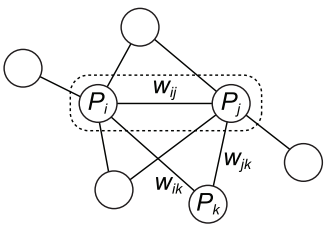

Figure 1. Example of a subgraph with an interacting protein pair and their neighboring proteins. P_i and P_j denote focusing interacting proteins shown in the dashed rectangle. P_k is a neighboring protein. w_{ij} denotes the weight of the interaction between P_i and P_j.
doi:10.1371/journal.pone.0065265.g001

discriminating heterodimeric complexes. Hence, we introduce the maximum of differences between the neighboring weights denoted by (F5).

For prediction of complexes, biological knowledge for proteins is helpful. We use protein domains that are parts of proteins known as structural and functional units. Ozawa et al. introduced the domain structural constraint that one domain interacts with at most one other domain for verifying protein complexes [20]. The constraint excludes extra proteins from a set of proteins that is a candidate complex by validating possible interactions between domains. This means that extra domains cause interactions with other proteins and the actual number of proteins contained in the complex may be greater than that in the candidate set of proteins. Since two proteins with small numbers of domains tend to form a heterodimeric complex, we introduce the maximum of the numbers of domains contained in P_i and P_j denoted by (F6). In contrast, we introduce the minimum of the numbers of domains contained in P_i and P_j denoted by (F7) because proteins with large numbers of domains tend to form complexes with large sizes.

Domain Composition Kernel

In the previous section, we introduced several feature space mappings from an example, that is, a pair of proteins. Kernel functions can incorporate prior knowledge. If a set of proteins has the same composition of domains as a known complex, it is highly expected that the set forms a complex. On the basis of this idea, we propose domain composition kernel for candidate complexes C_i and C_j with size n ($n=2$ in this paper), in which C_i and C_j are regarded as sets of proteins, $\{P_{i_1},\cdots,P_{i_n}\}$ and $\{P_{j_1},\cdots,P_{j_n}\}$, respectively. Then, we define equivalence $=_d$ between two proteins P_{i_k} and P_{j_l} as P_{i_k} consists of the same domains of P_{j_l}, where the number of each domain must also be the same between the proteins. Furthermore, we define equivalence $=_c$ between two sets of proteins C_i and C_j using $=_d$ by

$$C_i =_c C_j \Leftrightarrow \exists\sigma\in\mathfrak{S}_n \forall k (P_{i_k} =_d P_{j_{\sigma(k)}}), \tag{1}$$

where $\mathfrak{S}_n$ denotes the symmetric group of degree n on the set $\{1,\cdots,n\}$ (σ is a permutation of $(1,\cdots,n)$). For example, in the case of $C_i=\{P_{i_1},P_{i_2}\}$ and $C_j=\{P_{j_1},P_{j_2}\}$, $C_i=_c C_j$ if $P_{i_1}=_d P_{j_1}$ and $P_{i_2}=_d P_{j_2}$ or $P_{i_1}=_d P_{j_2}$ and $P_{i_2}=_d P_{j_1}$, whereas it is not necessary that $P_{i_1}=_d P_{i_2}=_d P_{j_1}=_d P_{j_2}$.

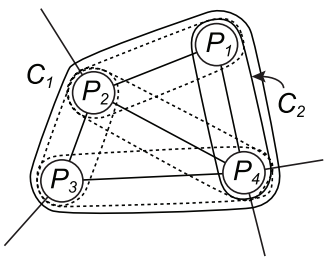

Figure 2. Illustration of the selection of negative examples from complexes with size more than two. Complex C_1 consists of four proteins $P_1,\cdots,P_4$, whereas heterodimeric complex C_2 consists of P_1 and P_4. Edges represent protein-protein interactions. According to this figure, four sets of two proteins, $\{P_1,P_2\}$, $\{P_2,P_3\}$, $\{P_2,P_4\}$, and $\{P_3,P_4\}$ are selected as negative examples. The set of two proteins $\{P_1,P_4\}$ is removed from the dataset. Each pair of two proteins surrounded by a dashed curve corresponds to a negative example.
doi:10.1371/journal.pone.0065265.g002

Then, we propose domain composition kernel K_c by

$$K_c(C_i,C_j)=\delta(C_i=_c C_j), \tag{2}$$

where $\delta(T)=1$ if T holds, otherwise 0. It should be noted that our kernel is different from pairwise kernels for protein pairs proposed in [21]. Their kernel is defined as $K_p(\{P_{i_1},P_{i_2}\},\{P_{j_1},P_{j_2}\})=K'_p(P_{i_1},P_{j_1})K'_p(P_{i_2},P_{j_2})+K'_p(P_{i_1},P_{j_2})K'_p(P_{i_2},P_{j_1})$ for predicting protein-protein interactions, where $K'_p(\cdot,\cdot)$ is called 'genomic kernel' and operates on individual genes or proteins. In the case of $C_i=_c C_j$, that is, $K_c=1$, $K_p=2$ if $P_{i_1}=_d P_{i_2}=_d P_{j_1}=_d P_{j_2}$, otherwise $K_p=1$, where $K'_p(P_i,P_j)=\delta(P_i=_d P_j)$. In addition, their pairwise kernels allow extra domains in a candidate complex because the domains do not prevent two proteins to interact with each other.

We can prove that $K_c(\cdot,\cdot)$ is a kernel.

Theorem 1 $K_c(\cdot,\cdot)$ defined by Eq. (2) is a positive semidefinite kernel.

Proof) We show that the Gram matrix $\boldsymbol{K}$ for a set of candidate complexes $C=\{C_1,\cdots,C_m\}$ is positive semidefinite. The binary relation $=_c$ on the candidate set is an equivalence relation because for all $C_i,C_j,C_k\in C$, $C_i=_c C_i$ (reflexivity), if $C_i=_c C_j$ then $C_j=_c C_i$ (symmetry), if $C_i=_c C_j$ and $C_j=_c C_k$ then $C_i=_c C_k$ (transitivity). Then, the relation $=_c$ partitions C into $S_1,\cdots,S_l$, and we have for any vector $\boldsymbol{x}=(x_1,\cdots,x_m)^T\in\boldsymbol{R}^m$

$$\boldsymbol{x}^T\boldsymbol{K}\boldsymbol{x}=\sum_{i=1}^{m}\sum_{j=1}^{m}K_{ij}x_i x_j, \tag{3}$$

$$=\sum_{i=1}^{l}\left(\sum_{C_j\in S_i}x_j\right)^2\geq 0. \tag{4}$$

It should be noted that $K_{ij}=K_c(C_i,C_j)=1$ if C_i and C_j are classified in the same set, otherwise $K_{ij}=0$. Consequently, $\boldsymbol{K}$ is positive semidefinite, and $K_c(\cdot,\cdot)$ is a valid kernel. □.

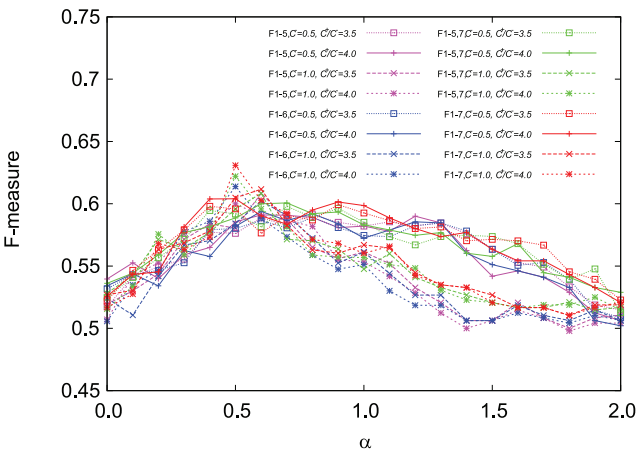

Figure 3. Result on the average F-measures using four sets of features and the domain composition kernel with $\alpha=0.0,0.1\cdots,2.0$**.** C-SVC was employed with regularization parameters, $C^-=0.5,1.0$, $C^+/C^-=3.5,4.0$. As sets of features, (F1–5), (F1–6), (F1–5,7), and (F1–7) shown in Table 1 were used.
doi:10.1371/journal.pone.0065265.g003

In addition, for the purpose of predicting whether or not two interacting proteins form a heterodimeric complex, we combine some feature space mapping ϕ in Table 1 with the domain composition kernel by

$$K(\phi(C_i),\phi(C_j))+\alpha K_c(C_i,C_j), \quad (5)$$

where $K(\cdot,\cdot)$ is any kernel for real-valued vectors, and α is a positive constant. In this paper, we use the linear kernel for K, that is, $K(\phi(C_i),\phi(C_j))=\langle\phi(C_i),\phi(C_j)\rangle$.

Computational Experiments

Data and Implementation

To perform computational experiments, we needed protein-protein interaction data with weights and protein complex data. We used the WI-PHI database [1] including 49607 protein pairs except self interactions as weighted protein-protein interaction data, where the actual file name was 'pro200600448_3_s.csv' at the supporting information web page of http://www.wiley-vch.de/contents/jc_2120/2007/pro200600448_s.html. The weights of interactions were calculated as follows. They constructed the literature-curated physical interaction (LCPH) dataset using several databases such as BioGRID [2], MINT [3], and BIND [4], and high-throughput yeast two-hybrid data by Ito [22] and Uetz [23]. To evaluate high-throughput data, they constructed a benchmark dataset having interactions supported by two independent methods from LCPH-LS, which was a low-throughput dataset in LCPH, and calculated a log-likelihood score (LLS) to each dataset except LCPH-LS. For each interaction, the weight was calculated by multiplying the socioaffinity (SA) indices [15] and the LLSs from different datasets, where the SA index measures the log-odds score of the number of times two proteins are observed to interact to the expected value from their frequency in the dataset.

To compare our method with the naive Bayes-based method proposed by Maruyama [11], we prepared the same dataset as in the paper [11] from CYC2008 protein complex database [12], which is available at http://wodaklab.org/cyc2008/resources/CYC2008_complex.tab. In the dataset, a positive example was restricted to a pair of proteins that is included as a PPI in WI-PHI and is not a proper subset of any other complex in CYC2008. Thus, we used 152 heterodimeric protein complexes contained in CYC2008 as positive examples, and selected 5345 negative examples from interacting protein pairs in the CYC2008 complexes with size more than two, where positive examples were excluded. Figure 2 shows an example of complexes C_1 and C_2 consisting of four proteins $P_1,\cdots,P_4$ and two proteins P_1 and P_4, respectively. According to this figure, four sets of two proteins, $\{P_1,P_2\}$, $\{P_2,P_3\}$, $\{P_2,P_4\}$, and $\{P_3,P_4\}$ are selected as negative examples, where each interaction between two proteins is confirmed to be included in WI-PHI. The set of two proteins $\{P_1,P_4\}$ is removed from the dataset. Since negative examples selected in this way are more difficult to be correctly predicted than randomly selected ones, this dataset is considered to be useful for the evaluation.

C-Support Vector Classification (C-SVC) for unbalanced data. Since the numbers of positive and negative examples of the dataset used in this paper were very unbalanced, we used the extension of C-Support Vector Classification (C-SVC) described in [24,25]. The extended C-SVC solves the following optimization problem given input feature vectors $\mathbf{x}_i$ and the corresponding classes $y_i\in\{+1,-1\}$.

$$\begin{aligned} \min \quad & \frac{1}{2}||\mathrm{w}||^2+C^+\sum_{y_i=+1}\xi_i+C^-\sum_{y_i=-1}\xi_i \\ \text{subject to} \quad & \forall i\ y_i(\mathrm{w}^T\cdot\mathrm{x}_i+b)\geq 1-\xi_i \\ & \forall i\ \xi_i\geq 0 \end{aligned}$$

where C^+ and C^- are regularization parameters for positive and negative classes, respectively, and in the usual C-SVC, $C^+=C^-$.

We used 'libsvm' (version 3.11) [26] as an implementation of C-SVC for unbalanced data.

Performance measure. To evaluate the performance of our method, we used precision, recall and F-measure, which are defined by

$$\text{precision}=\frac{TP}{TP+FP}, \quad (6)$$

Table 2. Result on the average precision, recall, and F-measure using our features and domain composition kernel in the best average F-measure case for each set of features.

method	features	α	C^-	C^+/C^-	precision	recall	F-measure
Our combination kernel	F1–5	0.6	0.7	4.0	0.586	0.659	0.620
	F1–6	0.7	0.8	3.5	0.566	0.677	0.616
	F1–5,7	0.6	0.7	4.0	0.592	0.667	0.627
	F1–7	0.5	1.0	4.0	0.618	0.644	**0.631**
naive Bayes [11]	B1, B2:CC	–			0.24	0.44	0.31
	B1–6	–			0.17	0.65	0.27
MCL [7]	–				0.017	0.023	0.020
MCODE [8]	–				0	0	–
RRW [18]	–				0.030	0.32	0.055
NWE [19]	–				0.035	0.33	0.063

As sets of features, (F1–5), (F1–6), (F1–5,7), and (F1–7) shown in Table 1 were used. The results by the naive Bayes-based method [11], MCL [7], MCODE [28], RRW [18], and NWE [19] are also shown, where the experiments for these methods were performed by [11]. (B1), (B2:CC),, (B6) indicate the features by [11] (shown also in Table 3).
doi:10.1371/journal.pone.0065265.t002

Table 3. Feature space mapping from two interacting proteins P_i, P_j in the naive Bayes-based method [11].

(B1)	$w_{ij} - \max\left\{ \max_{\{k\|(i,k)\in E, k\neq j\}} w_{ik}, \max_{\{k\|(j,k)\in E, k\neq i\}} w_{jk} \right\}$
(B2:X)	$w_{ij}^{GO:X} - \max\left\{ \max_{\{k\|(i,k)\in E, k\neq j\}} w_{ik}^{GO:X}, \max_{\{k\|(j,k)\in E, k\neq i\}} w_{jk}^{GO:X} \right\}$, where X represents
	an ontology among biological process (BP), cellular component (CC) and molecular
	function (MF) of Gene Ontology [27], and is also regarded to be the set of the terms;
	$w_{ij}^{GO:X} = -\|C_{ij}^X\| \log\left(\frac{\min_{t\in C_{ij}^X}\|S_t\|}{\max_{t\in X}\|S_t\|}\right)$, where C_{ij}^X is the set of all terms in X annotating
	both P_i and P_j, and S_t is the set of proteins annotated by term t.
(B3)	$r_{ij} - \max\left\{ \max_{\{k\|(i,k)\in E, k\neq j\}} r_{ik}, \max_{\{k\|(j,k)\in E, k\neq i\}} r_{jk} \right\}$, where $r_{ij} = \frac{\pi(i\to j)+\pi(j\to i)}{2}$ and
	$\pi(i\to j)$ is the stationary probability from P_i to P_j by a random walk with restarts
	at P_i (RRW [18]).
(B4)	$w_{ij}^{Exp} - \max\left\{ \max_{\{k\|(i,k)\in E, k\neq j\}} w_{ik}^{Exp}, \max_{\{k\|(j,k)\in E, k\neq i\}} w_{jk}^{Exp} \right\}$, where w_{ij}^{Exp} is the Pearson
	correlation coefficient between the two genes producing P_i and P_j, respectively, over
	some gene expression profiles.
(B5)	$\|\{k\|w_{ik}\geq w_{ij},(i,k)\in E,k\neq j\}\|+\|\{k\|w_{kj}\geq w_{ij},(k,j)\in E,k\neq i\}\|$
(B6)	$\|\{k\|(i,k),(k,j)\in E,k\neq i,j\}\|$

doi:10.1371/journal.pone.0065265.t003

$$\text{recall} = \frac{TP}{TP+FN}, \tag{7}$$

$$\text{F-measure} = \frac{2\cdot \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}, \tag{8}$$

where TP, FP, and FN denote the numbers of true positive, false positive, and false negative examples, respectively. Precision means the rate of correctly predicted positive examples to examples predicted as positive, and recall means the rate of correctly predicted positive examples to all positive examples. For evaluation of binary predictors, it is not sufficient to calculate only either the precision or the recall, and thus we used F-measure of their harmonic mean.

Results

To evaluate our method, we used several sets of our proposed features, (F1–5), (F1–6), (F1–5,7), and (F1–7). For example, (F1–5)

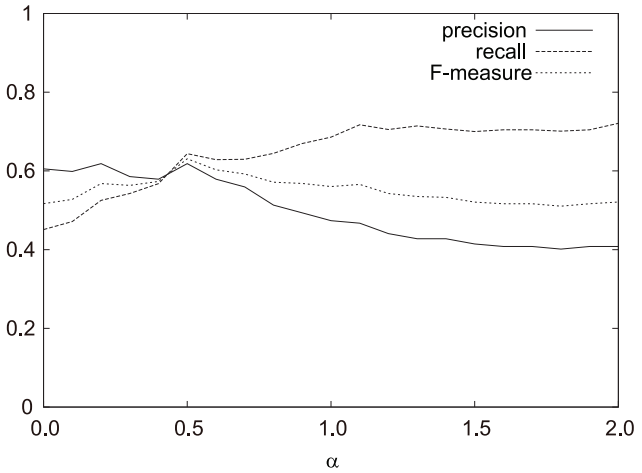

Figure 4. Result on the average precision, recall, and F-measure with varying $\alpha = 0.0, \cdots, 2.0$ in the best case using features (F1–7).
doi:10.1371/journal.pone.0065265.g004

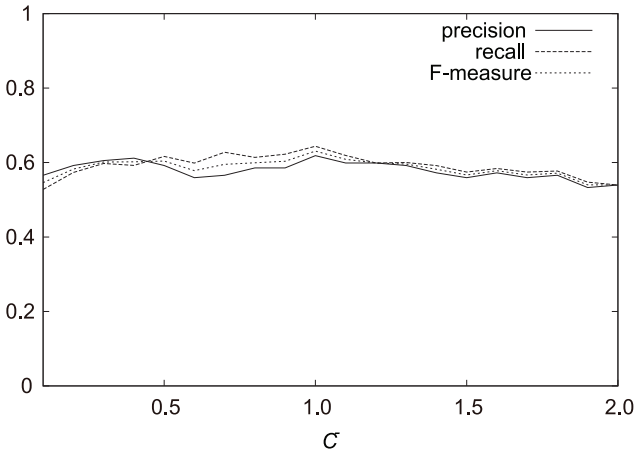

Figure 5. Result on the average precision, recall, and F-measure with varying $C^- = 0.1, \cdots, 2.0$ in the best case using features (F1–7).
doi:10.1371/journal.pone.0065265.g005

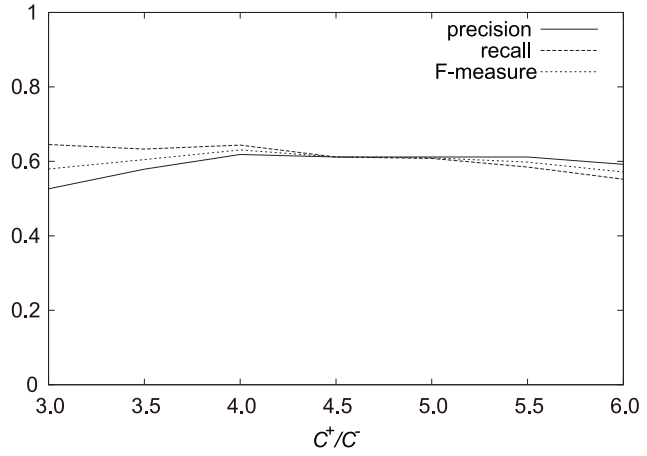

Figure 6. Result on the average precision, recall, and F-measure with varying $C^+/C^- = 3.0, \cdots, 6.0$ in the best case using features (F1–7).
doi:10.1371/journal.pone.0065265.g006

means that we use a feature vector consisting of five values calculated by (F1), (F2), $\cdots$, (F5) as shown in Table 1. Then, we calculated the combination kernel with the domain composition kernel as shown in Eq.(5), and employed C-SVC with varying mixing parameter $\alpha = 0.0, 0.1, \cdots, 2.0$ and regularization parameters $C^- = 0.1, 0.2, \cdots, 2.0$, $C^+/C^- = 3.0, 3.5, \cdots, 6.0$. For each case, we performed 10-fold cross-validation using our combination kernel, and took the average of precision, recall, and F-measure in the same way as in [11].

Figure 3 shows the results on the average F-measures using four sets of features, (F1–5), (F1–6), (F1–5,7), (F1–7), and the domain composition kernel for the cases of $\alpha = 0.0, 0.1 \cdots, 2.0$, $C^- = 0.5, 1.0$, $C^+/C^- = 3.5, 4.0$ (see Fig. S1 for more cases of $C^- = 0.1, 0.5, 1.0, 1.5, 2.0$ and $C^+/C^- = 3.0, 3.5, \cdots, 6.0$). We can see from these figures that the average F-measures during $0.5 \leq \alpha \leq 1.0$ were about 0.5 to 0.6 and were better than that of $\alpha = 0.0$ in each case. It means that the domain composition kernel enhanced the prediction accuracy comparing with only features. Furthermore, features (F1–7) tended to have better average F-measures than other sets of features.

Table 2 shows the results on the average precision, recall, and F-measure using our features and domain composition kernel in the best average F-measures case for each set of features. It also shows the results by the naive Bayes-based method [11], which is the best existing method for heterodimeric complex prediction, MCL [7], MCODE [8], RRW [18], and NWE [19]. (B1), (B2:CC), …, (B6) indicate the features used in the naive Bayes-based method (shown also in Table 3). These existing methods were executed using default parameters except the option of the minimum size of predicted complexes, which was set to be two if possible. For sets of features (F1–5), (F1–6), (F1–5,7), and (F1–7), the average F-measures in the cases of $(\alpha, C^-, C^+/C^-) = (0.6, 0.7, 4.0)$, $(0.7, 0.8, 3.5)$, $(0.6, 0.7, 4.0)$, and $(0.5, 1.0, 4.0)$ were best, respectively. In particular, the average F-measure for (F1–7) using $(\alpha, C^-, C^+/C^-) = (0.5, 1.0, 4.0)$ was best among all the cases, and was much better than that by the naive Bayes-based method. We investigated which feature most contributed to the prediction accuracy. The discriminant function for SVM with linear kernel can be represented as $f(x) = w^T + b$. Here we suppose that elements $w_1, \cdots, w_7$ of $\boldsymbol{w}$ are the coefficients of the corresponding features (F1),(F7), respectively. If each element of $\boldsymbol{x}$ is normalized, it can be considered that features with the largest absolute value of w_i are effective for the discrimination in the seven features. We calculated the coefficients and averages of the feature values using $(C^-, C^+/C^-) = (1.0, 4.0)$ and the dataset with 152 positive and 5345 negative examples. Thus, we had the coefficients $w = (0.049, 0.0052, 0.18, -0.063, -0.017, -0.066, 0.32)^T$, $b = -2.40$, and the averages $\bar{x} = (27.4, 56.5, 6.7, 33.2, 31.1, 1.8, 1.1)^T$. Then, $(w_i \cdot \bar{x}_i) = (1.35, 0.29, 1.18, -2.09, -0.54, -0.12, 0.35)^T$, and it was (F4),(F1),(F3),(F5),(F7),(F2),(F6) in descending order of $|w_i \cdot \bar{x}_i|$. We can see that (F4) was most effective, and worked on the discrimination negatively, whereas (F6) was least effective, in fact, the decrease of the average F-measure by removal of (F6) from (F1–7) was small as shown in Table 2. It should be noted that this result does not necessarily mean that supervised methods such as the naive Bayes-based method and our proposed method are always better than unsupervised methods such as MCL and MCODE because unsupervised methods were evaluated using the whole PPI data whereas supervised methods were trained and evaluated via cross validation using a part of PPI data. Therefore, unsupervised methods may work better in other situations.

Figures 4, 5, and 6 show the results on the average precision, recall, and F-measure with varying α, C^-, and C^+/C^-, respectively, in the case of $(\alpha, C^-, C^+/C^-) = (0.5, 1.0, 4.0)$ using features (F1-7). We can see that in the examined range, the average F-measures did not largely fluctuated.

In addition, we performed another experiment to validate our method for the rest PPIs, that is, we used 152 positive and 5345 negative examples as training data, and used the rest, 44110 examples as test data. Then, we obtained the prediction accuracy of 98.7% (43554/44110) using the combination kernel with (F1–7) and $(\alpha, C^-, C^+/C^-) = (0.5, 1.0, 4.0)$. These results suggest that our proposed kernel successfully predicted heterodimeric protein complexes and outperforms the naive Bayes-based method.

Conclusions

We proposed several feature space mappings using weights of protein-protein interactions for predicting heterodimeric protein complexes. In addition, we proposed the domain composition kernel based on the idea that two proteins having the same composition of domains as a heterodimeric protein complex would also form a heterodimer, and proved that the domain composition kernel is actually a kernel function. To validate our proposed method, we performed ten-fold cross-validation computational experiments for the combination kernel of the domain composition kernel with the linear kernel using several sets of features. The results suggest that our proposed kernel considerably outperforms the naive Bayes-based method, which is the best existing method, even in the case using only feature space mappings (F1–5) from weights of protein-protein interactions, that is, (F6,7) was not used and the mixing parameter α is 0 although our proposed method is limited to prediction of heterodimeric protein complexes.

An important contribution in this paper is that we have shown that heterodimeric protein complexes are able to be successfully predicted using only information on weights of protein-protein interactions. Furthermore, we indicated that the use of protein domain information enhances the prediction accuracy.

There is some possibility to further improve the prediction accuracy. For instance, we can develop some kernels on protein domains using protein amino acid sequences and multiple sequence alignments. In addition, we can add new features based on other biological knowledge.

We used the C-SVC classifier, which is a variant of support vector machines, because the numbers of positive and negative examples were not balanced. It is interesting future work to

develop more robust methods against unbalanced data for classifying heterodimeric protein complexes.

Supporting Information

Figure S1 Result on the average F-measures using four sets of features and the domain composition kernel with $\alpha = 0.0, 0.1 \cdots, 2.0$. *C*-SVC was employed with regularization parameters, $C^- = 0.1, 0.5, 1.0, 1.5, 2.0$, $C^+/C^- = 3.0, 3.5, \cdots, 6.0$. As sets of features, (F1–5), (F1–6), (F1–5,7), and (F1–7) shown in Table 1 were used.
(EPS)

Figure S2 Result on the average F-measures using four sets of features and the domain composition kernel represented by Eq. (S1) with $\beta = 0.0, 0.1 \cdots, 1.0$. *C*-SVC was employed with regularization parameters, $C^- = 0.5, 1.0$, $C^+/C^- = 3.5, 4.0$. As sets of features, (F1–5), (F1–6), (F1–5,7), and (F1–7) were used.
(EPS)

Table S1 Result on the average precision, recall, and F-measure using our combination kernel represented by Eq. (S1) in the best average F-measure case for each set of features. As sets of features, (F1–5), (F1–6), (F1–5,7), and (F1–7) were used.
(PDF)

Text S1 Results on our kernel by another combination.
(PDF)

Author Contributions

Conceived and designed the experiments: MH OM TA. Performed the experiments: PR. Analyzed the data: PR MH. Contributed reagents/materials/analysis tools: PR MH. Wrote the paper: PR MH OM TA.

References

1. Kiemer L, Costa S, Ueffing M, Cesareni G (2007) WI-PHI: A weighted yeast interactome enriched for direct physical interactions. Proteomics 7: 932–943.
2. Stark C, Breitkreutz B, Reguly T, Boucher L, Breitkreutz A, et al. (2006) BioGRID: a general repository for interaction datasets. Nucleic Acids Research 34: D535–D539.
3. Zanzoni A, Montecchi-Palazzi L, Quondam M, Ausiello G, Helmer-Citterich M, et al. (2002) MINT: a Molecular INTeraction database. FEBS Letters 513: 135–140.
4. Alfarano C, Andrade C, Anthony K, Bahroos N, Bajec M, et al. (2005) The Biomolecular Interac- tion Network Database and related tools 2005 update. Nucleic Acids Research 33: D418–D424.
5. Sapkota A, Liu X, Zhao XM, Cao Y, Liu J, et al. (2011) DIPOS: database of interacting proteins in *Oryza sativa*. Molecular BioSystems 7: 2615–2621.
6. Zhao XM, Zhang XW, Tang WH, Chen L (2009) FPPI: *Fusarium graminearum* protein-protein interaction database. J Proteome Res 8: 4714–4721.
7. Enright A, Dongen SV, Ouzounis C (2002) An efficient algorithm for large-scale detection of protein families. Nucleic Acids Research 30: 1575–1584.
8. Bader GD, Hogue CW (2003) An automated method for finding molecular complexes in large protein interaction networks. BMC Bioinformatics 4: 2.
9. King A, Prulj N, Jurisica I (2004) Protein complex prediction via cost-based clustering. Bioinfor- matics 20: 3013–3020.
10. Chua H, Ning K, Sung WK, Leong H, Wong L (2008) Using indirect protein-protein interactions for protein complex prediction. Journal of Bioinformatics and Computational Biology 6: 435–466.
11. Maruyama O (2011) Heterodimeric protein complex identification. In: ACM Conference on Bioin- formatics, Computational Biology and Biomedicine 2011. 499–501.
12. Pu S, Wong J, Turner B, Cho E, Wodak S (2009) Up-to-date catalogues of yeast protein complexes. Nucleic Acids Research 37: 825–831.
13. Mewes HW, Amid C, Arnold R, Frishman D, Guldener U, et al. (2004) MIPS: analysis and anno- tation of proteins from whole genomes. Nucleic Acids Research 32: D41–D44.
14. Ho Y, Gruhler A, Heilbut A, Bader GD, Moore L, et al. (2002) Systematic identification of protein complexes in saccharomyces cerevisiae by mass spectrometry. Nature 415: 180–183.
15. Gavin AC, Aloy P, Grandi P, Krause R, Boesche M, et al. (2006) Proteome survey reveals modu- larity of the yeast cell machinery. Nature 440: 631–636.
16. Krogan NJ, Cagney G, Yu H, Zhong G, Guo X, et al. (2006) Global landscape of protein complexes in the yeast saccharomyces cerevisiae. Nature 440: 637–643.
17. Qi Y, Balem F, Faloutsos C, Klein-Seetharaman J, Bar-Joseph Z (2008) Protein complex identification by supervised graph local clustering. Bioinformatics 24: i250–i258.
18. Macropol K, Can T, Singh A (2009) Repeated random walks on genome-scale protein networks for local cluster discovery. BMC Bioinformatics 10: 283.
19. Maruyama O, Chihara A (2010) NWE: Node-weighted expansion for protein complex prediction using random walk distances. In: 2010 IEEE International Conference on Bioinformatics and Biomedicine (BIBM2010). 590–594.
20. Ozawa Y, Saito R, Fujimori S, Kashima H, Ishizaka M, et al. (2010) Protein complex prediction via verifying and reconstructing the topology of domain-domain interactions. BMC Bioinformatics 11: 350.
21. Ben-Hur A, Noble W (2005) Kernel methods for predicting protein-protein interactions. Bioinfor- matics 21: i38–i46.
22. Ito T, Chiba T, Ozawa R, Yoshida M, Hattori M, et al. (2001) A comprehensive two-hybrid analysis to explore the yeast protein interactome. Proc Natl Acad Sci USA 98: 4569–4574.
23. Uetz P, Giot L, Cagney G, Mansfield T, Judson R, et al. (2000) A comprehensive analysis of protein-protein interactions in Saccharomyces cerevisiae. Nature 403: 623–627.
24. Osuna E, Freund R, Girosi F (1997) Support vector machines: Training and applications. In: AI Memo 1602, Massachusetts Institute of Technology.
25. Vapnik V (1998) Statistical Learning Theory. Wiley-Interscience.
26. Chang CC, Lin CJ (2011) LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2: 27: 1–27: 27.
27. Gene Ontology Consortium (2008) The Gene Ontology project in 2008. Nucleic Acids Research 36: D440–D444.