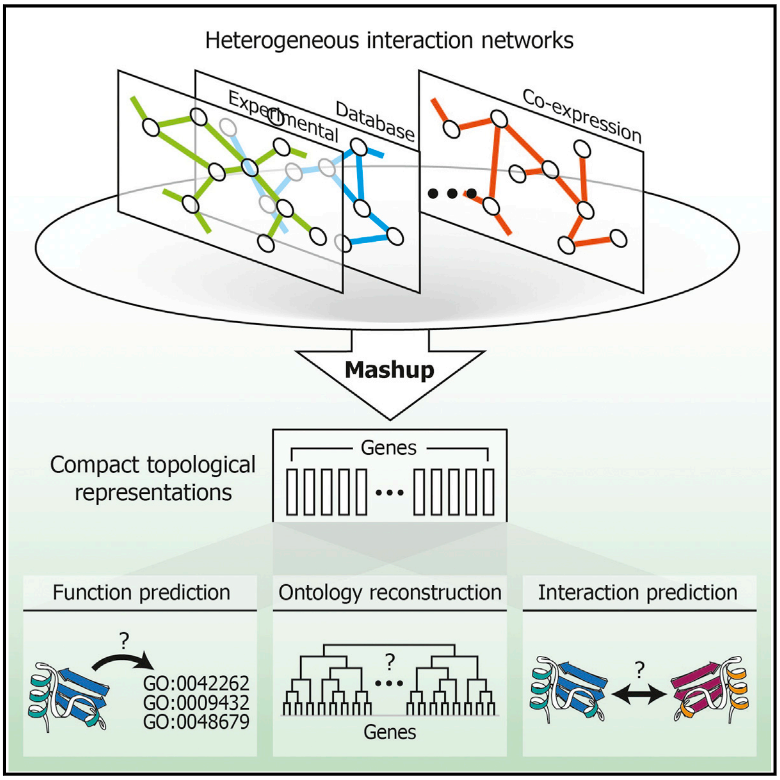
Cell Systems

Article

Compact Integration of Multi-Network Topology for Functional Analysis of Genes

Graphical Abstract



Highlights

- We learn compact features of topology from multiple heterogeneous networks
- Our features obtain state-of-the-art accuracy in diverse functional inference tasks
- Our method scales to many networks and can be broadly applied to network science

Authors

Hyunghoon Cho, Bonnie Berger, Jian Peng

Correspondence

bab@mit.edu (B.B.),
jianpeng@illinois.edu (J.P.)

In Brief

Mashup is a computational approach for integrating data across multiple networks by compactly representing the topological relationships between nodes.

Cho et al., 2016, Cell Systems *3*, 540–548
December 21, 2016
http://dx.doi.org/10.1016/j.cels.2016.10.017

OPEN ACCESS
CellPress

Compact Integration of Multi-Network Topology for Functional Analysis of Genes

Hyunghoon Cho,[1] Bonnie Berger,[1,2,*] and Jian Peng[1,3,4,*]
[1]Computer Science and Artificial Intelligence Laboratory, MIT, Cambridge, MA 02139, USA
[2]Department of Mathematics, MIT, Cambridge, MA 02139, USA
[3]Department of Computer Science, University of Illinois at Urbana-Champaign, Champaign, IL 61801, USA
[4]Lead Contact
*Correspondence: bab@mit.edu (B.B.), jianpeng@illinois.edu (J.P.)
http://dx.doi.org/10.1016/j.cels.2016.10.017

SUMMARY

The topological landscape of molecular or functional interaction networks provides a rich source of information for inferring functional patterns of genes or proteins. However, a pressing yet-unsolved challenge is how to combine multiple heterogeneous networks, each having different connectivity patterns, to achieve more accurate inference. Here, we describe the Mashup framework for scalable and robust network integration. In Mashup, the diffusion in each network is first analyzed to characterize the topological context of each node. Next, the high-dimensional topological patterns in individual networks are canonically represented using low-dimensional vectors, one per gene or protein. These vectors can then be plugged into off-the-shelf machine learning methods to derive functional insights about genes or proteins. We present tools based on Mashup that achieve state-of-the-art performance in three diverse functional inference tasks: protein function prediction, gene ontology reconstruction, and genetic interaction prediction. Mashup enables deeper insights into the structure of rapidly accumulating and diverse biological network data and can be broadly applied to other network science domains.

INTRODUCTION

Comprehensively understanding various functional aspects of genes or proteins, such as their involvement in a particular biological process, physical or genetic interactions, or disease association, is critical for both biological and translational medicine research. Since exhaustively characterizing genes or proteins through biological experiments is often intractable, systems-level integration of knowledge and computational hypothesis generation have garnered great interest in the field as effective ways to guide experiments (Berger et al., 2013).

With the advent of high-throughput experimental techniques, genome-scale interaction networks (also known as interactomes) have been an integral way of encapsulating information and have enabled approaches to extend and refine functional knowledge of genes and proteins (Yu et al., 2013). A key insight behind such approaches is that genes or proteins that are co-localized or have similar topological roles in the interaction networks are more likely to be functionally correlated. This insight allows us to infer properties of unknown proteins by transferring knowledge from similar genes and proteins that are better understood—a process known as "guilt by association."

An important challenge has been to develop principled approaches for integrating heterogeneous sources of information (e.g., physical binding, genetic interaction, co-expression, or co-evolution) from which different interaction networks can be constructed. Most previous work has focused on summarizing a collection of heterogeneous data into a single integrated network, which is typically obtained by combining the edges across different networks via Bayesian inference (Franceschini et al., 2013; Lee et al., 2011; Wong et al., 2015) or adaptive weighted averaging (Mostafavi et al., 2008). The resulting integrated network is provided as input to existing network-based inference methods, such as label propagation (Mostafavi et al., 2008) or graph-based clustering (Dutkowski et al., 2013), to derive functional insights from the data. However, a key limitation of such approaches is the substantial information loss incurred by projecting various datasets onto a single network representation. For instance, context-specific interaction patterns (e.g., tissue-specific gene modules) that are only present in certain datasets are likely to be obscured by edges from other data sources in the integrated network.

A naive approach for tackling this challenge would be to separately analyze the structure of each network and to concatenate the resulting network features (e.g., Cao et al., 2014; Milenković and Pržulj, 2008; Mostafavi et al., 2012) for each gene. However, this approach greatly increases the dimensionality of the feature space and often dilutes the signal in the data as a result. Noise in interaction networks based on high-throughput experiments further compounds this issue. Thus, it is imperative to develop integrative methods that can properly take advantage of the fine-grained topology of multiple heterogeneous networks while maintaining a low-dimensional feature space, thereby increasing robustness to noise and enhancing accuracy.

Here, we address this challenge by introducing an integrative framework, Mashup, for obtaining high-quality, compact topological feature representations of genes from one or more interaction networks constructed from heterogeneous data types. We incorporate the following conceptual advances into our

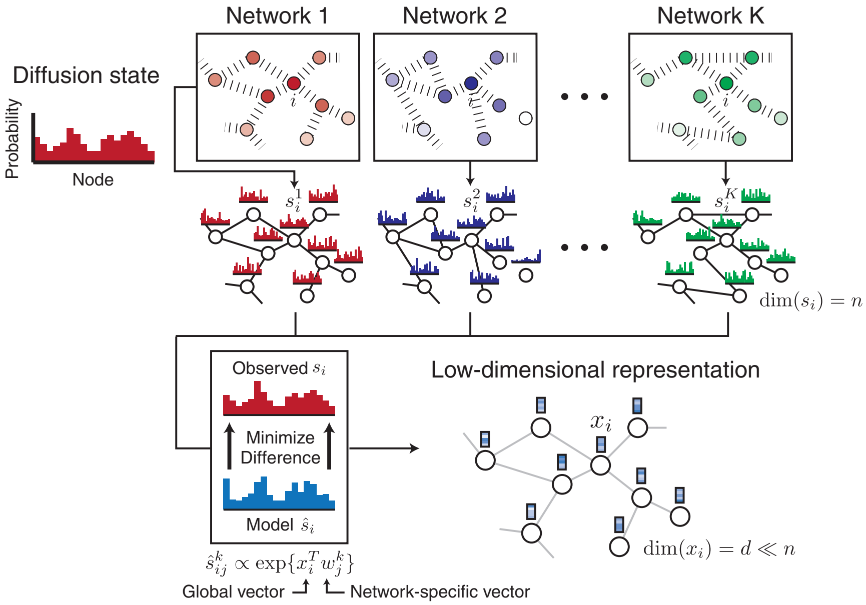


Figure 1. Overview of Mashup

Random walks with restart (RWR) are used to compute the diffusion state for each node in each individual network. Low-dimensional feature vectors describing the topological properties of each node are obtained by jointly minimizing the difference between the observed diffusion states and the parameterized-multinomial logistic distributions across all networks. The low-dimensional representation can be readily plugged into machine learning methods for functional inference.

framework: (1) Mashup takes full advantage of network-specific topology by analyzing the structure of each network separately before learning a canonical representation that best explains the topological patterns across all networks, and (2) Mashup decouples the dimensionality of feature representations from the data parameters (e.g., number of networks or genes), which allows it to cope with inherent noise in high-throughput data by obtaining *compact* representations that keep only the most explanatory features. By showing substantial improvements over the state-of-the-art methods in three distinct functional inference tasks—automated gene function annotation, gene ontology reconstruction, and genetic interaction prediction—we demonstrate Mashup's wide applicability and its potential to effectively decipher functional properties of genes from interactomes. Notably, Mashup easily scales to a large number of networks—a critical requirement for network-based methods to fully utilize the ever-growing repository of interactomes. We provide software for Mashup along with ready-to-use compact vector representations of genes learned from existing interactome datasets for researchers to apply to their own application domains (http://mashup.csail.mit.edu).

Mashup can in principle be used to simultaneously analyze any large networks in which guilt-by-association properties hold for more accurate knowledge discovery. Not only do the substantial improvements in accuracy and scalability promise to enable new workflows for biomedical practitioners (e.g., integration of single-cell data), but also the general framework for network integration that we introduce can be straightforwardly applied to network analysis problems outside of biology.

RESULTS

Overview of Mashup

The basic Mashup framework for heterogeneous network integration involves three steps (Figure 1). (1) Run a localized network diffusion process (e.g., random walks with restart [RWR; Tong et al., 2006]) on each network to obtain a distribution for each node, which captures its relevance to all other nodes in the network. Similar to the widely used PageRank algorithm (Page et al., 1999) in web and social network analysis, this step characterizes the topological context of each gene in a network, taking the global connectivity patterns into account. (2) Approximate each of these distributions by constructing a model, parameterized by low-dimensional feature vectors for each node; these feature vectors are obtained by minimizing the difference between the model distribution and diffusion distributions for all networks simultaneously. Akin to Principal Component Analysis (PCA), which reveals the internal low-dimensional linear structure of the data that best explains the variance, Mashup computes a low-dimensional vector-space representation for all nodes such that the diffusion or the connectivity patterns in the networks can be best explained. (3) Use the learned representations as input features for a wide range of network-based functional inference tasks. A more detailed description of Mashup is provided in Method Details.

Improved Gene Function Prediction

Automated annotation of gene function, the goal of which is to assign a poorly understood gene to the correct functional categories in an annotation database, is considered one of the most important and challenging problems of the post-genomic era (Radivojac et al., 2013). Many solutions based on high-throughput experimental data have been proposed in the past decade, each exploiting different types of information, including amino acid sequence (Clark and Radivojac, 2011), genomic context (Enault et al., 2005), evolutionary relationships (Gaudet et al., 2011), protein structure (Pal and Eisenberg, 2005), and gene expression (Huttenhower et al., 2006). Here, we focus on the use of protein-protein interaction (PPI) networks, where we pursue the intuition that the topological role of a gene in interaction networks is correlated with its biological function.

Existing approaches for integrating multiple networks for function prediction have largely focused on combining the networks into a single representative network to be used for prediction. GeneMANIA (Mostafavi and Morris, 2010; Mostafavi et al., 2008) is a state-of-the-art function prediction server that uses a label propagation algorithm on an averaged network, whose mixing weights are optimized for each functional category.

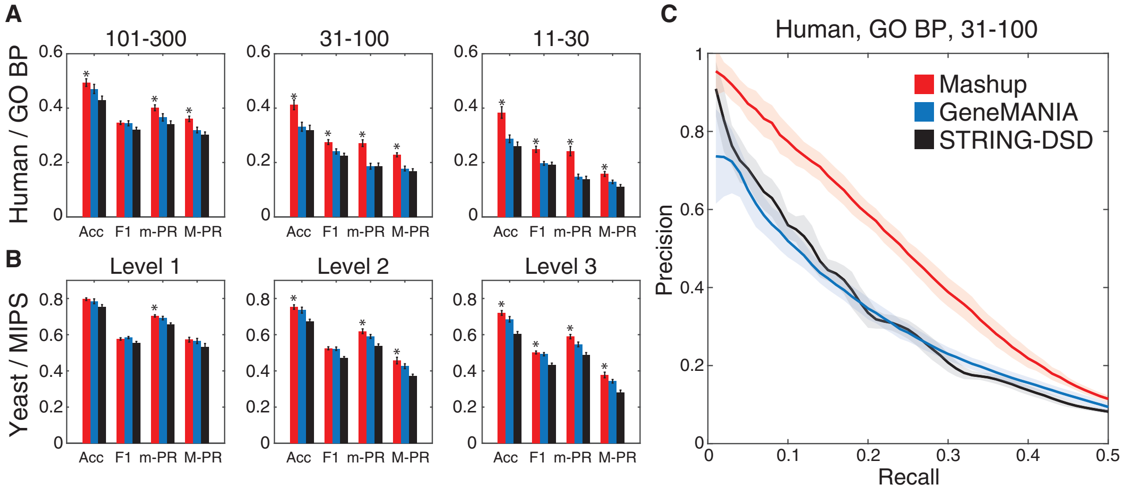


Figure 2. Mashup Improves Gene Function Prediction Performance in Human and Yeast

We performed 5-fold cross-validation to compare the function prediction performance of Mashup to other state-of-the-art network integration methods, GeneMANIA and STRING's Bayesian integration followed by a diffusion-based function prediction method DSD (STRING-DSD) in (A) human and (B) yeast. A precision-recall curve for each method is shown (C). Additional figures, including the results on molecular function (MF) and cellular component (CC) ontologies in human and further comparisons to other integration approaches, are provided in Figures S1, S2, and S3. Performance is measured by the fraction of top predictions correctly labeled (Acc), harmonic mean of precision and recall when the top three predictions are assigned to each gene (F1), and the area under the precision recall curve summarized over all labels, both under the micro-averaging (m-PR) and macro-averaging (M-PR) schemes. Results are summarized over ten trials (SD shown as error bars), and asterisks represent where Mashup's improvement over GeneMANIA is significant (one-sided rank-sum p value <0.01). Overall, Mashup achieves substantially greater predictive performance over previous methods.

Another standard approach for network integration, adopted by the large public PPI network database STRING (Franceschini et al., 2013), is to use Bayesian inference to combine edges across multiple networks. STRING's resulting integrated network can be used with single-network function prediction methods, such as diffusion state distance (DSD; Cao et al., 2014), a state-of-the-art diffusion-based method that uses RWR to characterize the local topology of each gene and assigns functions by majority vote based on a set of genes with most similar diffusion patterns.

We found that Mashup-based function prediction substantially outperforms these state-of-the-art integrative methods in assigning a previously unseen gene to its known functional categories in a cross-validation experiment on real datasets from yeast and human (Figure 2). We observed clear improvements for both the yeast and human datasets at different annotation levels of the Munich Information Center for Protein Sequences (MIPS) (Ruepp et al., 2004) and the Gene Ontology database (GO; Ashburner et al., 2000) hierarchies, respectively. For example, top predictions based on Mashup correctly assigned 38.4% of genes (on average) to their functional categories, in contrast to 28.7% for GeneMANIA and 25.9% for DSD with STRING integration (referred to as STRING-DSD), with respect to human Biological Process (BP) GO terms with highest specificity (11–30 genes). An exception to the general improvement was the top layer (level 1) of the yeast dataset, for which Mashup performed comparably to GeneMANIA. This finding is likely due to the relative completeness of yeast interactomes and the fact that the top layer contains the largest functional terms that are easiest to predict, leaving little room for improvement. Moreover, Mashup's improvement is consistent over a wide range of parameters in our framework, which includes the dimensionality of our learned representation and the restart probability of RWR (Figure S4). To enable function prediction with Mashup, we used a support vector machine (SVM) classifier for each functional category with Mashup's compact topological representations as input features (Method Details).

Mashup's accuracy improvement can be partially attributed to the fact that separately analyzing the structure of each individual network uncovers fine-grained topological patterns that are difficult to identify in the combined network where different edge types are not distinguished. For instance, we noted that many genes' most topologically similar gene, based on Mashup's integrated features, is not a direct neighbor in any of the networks, but rather a gene indirectly connected by numerous paths that go through different intermediary nodes in different networks. Such indirect, but *consistent* associations are often outweighed by direct neighbors if analyzed based on a single combined network, even if the direct connection exists only in a narrow context (few networks). Further inspection revealed that many of these top, indirect associations newly identified by Mashup in fact correspond to paralogous genes, suggesting that such patterns reflect coherent biological functions (Table S1).

Another important factor in Mashup's enhanced accuracy is the compactness of its feature representations, which helps tease functionally relevant topological patterns apart from noise in the data. To assess this aspect in isolation, we applied Mashup to individual networks without integration. We still observed significantly better (rank-sum p value <0.01) prediction performance as compared to the single-network method DSD on all but one network (Figure 3). As additional evidence, we observed that even a favorably modified DSD, which uses log-transformed diffusion states as features to train SVM classifiers to closely approximate Mashup without dimensionality

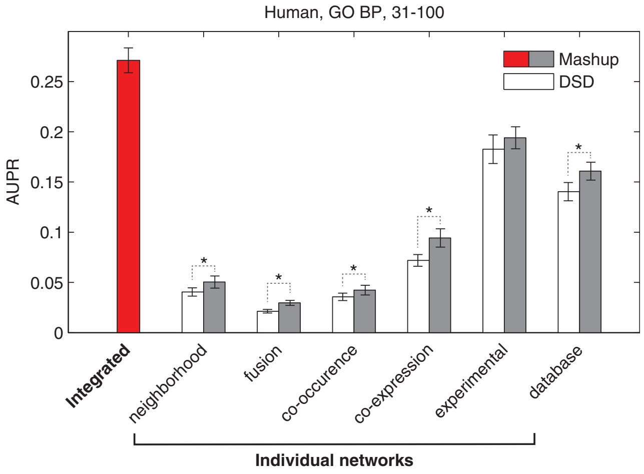


Figure 3. Integrating Multiple Networks Outperforms Individual Networks in Gene Function Prediction

We compared Mashup's 5-fold cross-validation performance, measured by micro-averaged area under the precision-recall curve (AUPR); performance on each individual network in STRING (gray shaded) is compared to using all networks simultaneously (Integrated, red shaded). The results of applying a diffusion-based, single network method, DSD, to each network type is also shown (white shaded). Asterisks represent individual networks where Mashup outperformed DSD (one-sided rank-sum p value <0.01). Results are summarized over ten trials (SD shown as error bars).

reduction, still achieved significantly lower accuracy than Mashup, which further corroborates the importance of the compactness of Mashup representations (Figure S6). Furthermore, randomly perturbing the network structure led to smaller changes in pairwise topological similarities between genes for Mashup features, compared to high-dimensional diffusion states used by DSD (Figure S7). This result demonstrates Mashup's greater robustness to noise. We would like to emphasize that integrating all networks from STRING results in higher function prediction performance than any single network alone (Figure 3), which underscores the significance of integrating various types of data sources for understanding the functional roles of genes or proteins.

Taken together, these results suggest that the key advances of Mashup—simultaneously capturing the patterns of multiple interaction networks by learning compact, canonical representations of topology—lead to substantially more accurate prediction of gene function than previous approaches.

Further comparisons to other data integration methods that previously have not been systematically evaluated for the task of function prediction are provided in Figure S1. In particular, we compared Mashup to a recently proposed matrix factorization-based approach, Collective-Matrix Factorization (CMF; Žitnik et al., 2015; Žitnik and Zupan, 2015), which views heterogeneous data matrices as relations between different object types that can be approximated via a low-lank factorization. While straightforward CMF has limited use of network data as additional constraints on the parameters to be learned, we considered a favorably modified CMF that directly factorizes the network data (i.e., more similar to Mashup) and found that Mashup significantly outperforms this approach as well (Figure S1).

More Precise Reconstruction of Gene Ontology

In addition to refining our functional knowledge of proteins via automatic function annotation, which assumes a predetermined set of functional categories, molecular networks can be used to guide the identification of functional categories and their hierarchical organization—widely known as "gene ontology." Building an entire ontology based on only high-throughput interactome data—an approach recently pioneered by Dutkowski et al. (2013)—circumvents the inconsistencies and biases that are typically introduced by the manual curation process underlying existing ontology databases (e.g., Gene Ontology [GO] database [Ashburner et al., 2000]). Therefore, such unbiased approaches can produce valuable hypotheses for enhancing and expanding existing ontologies.

Dutkowski et al. (2013) used a graph-based agglomerative clustering algorithm (Park and Bader, 2011) to extract a hierarchy of gene clusters from an interaction network, where each cluster is viewed as a putative functional category. The resulting data-driven ontology, called NeXO, was then provided as input to an ontology alignment algorithm developed by the same researchers to show substantial overlap with the GO database. More recently, a new algorithm based on maximal clique detection, named CliXO (Kramer et al., 2014), was proposed as an alternative approach that can better handle weighted interaction networks. Motivated by the observation that Mashup's integrated topological features are highly predictive of gene function, we set out to test whether clustering Mashup features in lieu of the original input networks would result in more accurate gene ontology than NeXO and CliXO. Both methods, unlike Mashup, take a single combined network as input, which obscures the fine-grained topological patterns that are specific to individual networks.

We first extracted compact topological representations with Mashup from the same set of four binary yeast PPI networks used by Dutkowski et al. (2013), which consists of a physical interaction network, a genetic interaction network, a co-expression network, and a functional association network from YeastNet (Kim et al., 2013). While Dutkowski et al. (2013) simply took the union of all edges to construct a combined network for clustering, we used topological features from Mashup's integration to construct a gene ontology via a standard hierarchical agglomerative clustering algorithm. The clustering was followed by a post-processing step analogous to NeXO's in order to introduce multi-way joins and multiple parents, which are common in real ontologies (Method Details). With our Mashup-based ontology, we achieved substantially better agreement with GO than NeXO (in F1 score, which measures harmonic mean of precision and recall) for molecular function (MF) and cellular component (CC) ontologies, and comparable performance for biological process (BP) (Figure 4A). Overall, Mashup achieved a combined alignment score of 0.33 (geometric mean of F1 scores in three ontology categories), which was significantly higher than NeXO's (0.24). To compare with CliXO (for weighted interaction networks), we applied Mashup to six weighted PPI networks, excluding text mining, from STRING (Franceschini et al., 2013) and similarly constructed an ontology via hierarchical clustering. For CliXO, we uniformly combined the networks by Bayesian integration (following STRING's approach) to construct a single integrated network as input. Even with an

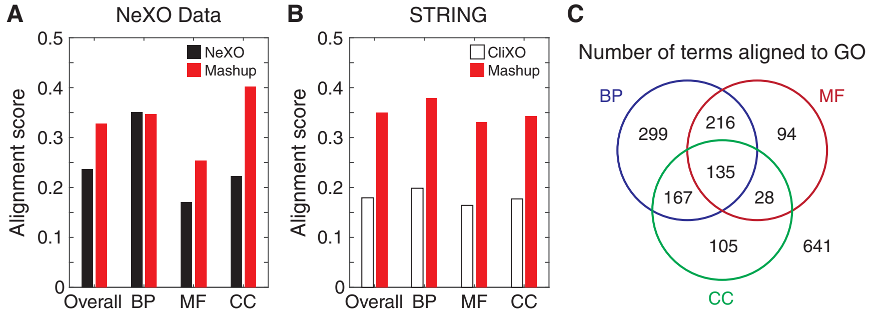


Figure 4. Mashup Improves Network-Based Gene Ontology Reconstruction

(A and B) Mashup features extracted from (A) the network data used to generate NeXO or (B) STRING networks were hierarchically clustered to generate an ontology, which is aligned using the same algorithm as NeXO/CliXO to biological process (BP), molecular function (MF), and cellular component (CC) ontologies in the GO database. We compared the alignment quality of Mashup-based ontologies to NeXO and CliXO on the respective datasets. Following previous work (Dutkowski et al., 2013), we measured the alignment quality as the harmonic mean of the fraction of terms in the reconstructed ontology and the fraction of terms in GO that are aligned. The overall score was calculated as the geometric mean of the three scores for different GO types.

(C) Breakdown of the number of terms in Mashup-based ontology using STRING networks aligned to GO (at FDR = 10%).

optimized parameter setting for CliXO (Method Details), Mashup-based ontology achieved substantially higher alignment scores than CliXO in all three ontology categories: Mashup had a combined score of 0.35, whereas CliXO, 0.18 (Figure 4B). Furthermore, we observed similar improvement for Mashup on the YeastNet networks (Kim et al., 2013), the original dataset used by Kramer et al. (2014) to evaluate CliXO (Figure S8).

These results demonstrate that Mashup's integration of topological features enable more precise identification of functionally coherent gene sets than the state-of-the-art approaches, which rely on a single combined network where network-specific topological information is obscured. Note also that Mashup allows the use of a simple off-the-shelf clustering algorithm through its convenient vector representation of topology.

Improved Prediction of Genetic Interaction and Drug Efficacy

A critical step toward attaining a thorough understanding of how genes carry out their biological function in a cell is to tease apart their sophisticated interplay with other genes or proteins. Synthetic lethality (SL) and synthetic dosage lethality (SDL) describe a particular type of interaction between genes where an otherwise non-essential gene becomes *essential* (i.e., its deletion reduces cell viability) given the deletion (SL) or overexpression (SDL) of another gene. SL interactions can reveal inherent redundancy in the genetic program, and SDL interactions, dosage dependence of gene products. There has been great interest in identifying SL or SDL interactions due to their clinical significance; these interactions can lead to the discovery of novel drugs for targeted therapies, where SL or SDL interaction partners of genes selectively deleted or overexpressed in disease cells are targeted (Chan and Giaccia, 2011). However, experimentally interrogating the presence of an interaction between every pair of genes is infeasible, and thus it is essential to develop computational approaches for predicting candidate interactions with high accuracy.

Several prediction methods have focused on the use of PPI networks either exclusively (Paladugu et al., 2008) or in conjunction with other types of information, such as gene expression or functional annotation (Pandey et al., 2010; Wong et al., 2004). The key insight behind these approaches is that observing a genetic interaction between genes A and B increases the likelihood of an interaction between A and other genes that are functionally similar to B. This kind of information transfer can be effectively derived from the topology of interactomes, as we demonstrated in the above applications of Mashup. Notably, Paladugu et al. (2008) used a manually curated list of conventional graph theoretic measures (e.g., degree, closeness, betweenness centralities) to train support vector machine (SVM) classifiers and demonstrated that analyzing the topology of a PPI network alone can be effective for predicting genetic interactions.

Here, we asked whether the compact topological representation learned by Mashup can be used to further improve prediction of genetic interactions. To this end, we adopted the same prediction framework used by Paladugu et al. (2008) and measured, via cross-validation, the impact of substituting Mashup's topological representations for their curated topological features (Method Details). We observed that Mashup's compact representation consistently outperforms manually curated topological features (referred to as graph-theoretic measures [GTM]) for predicting both SL and SDL interactions in a real human dataset (Figure 5A). Mashup achieved an average area under the precision recall curve (AUPR) of 0.59 for SL and 0.51 for SDL, whereas GTM achieved 0.44 and 0.39, respectively. Mashup's performance was highly consistent across a wide range of choices for the dimensionality of our learned representation (Figure S5). Furthermore, Mashup achieved substantially better accuracy than a recent approach that uses known GO annotations of each gene pair as input features for random forest classifiers (Yu et al., 2016; referred to as Ontotype), which was shown to achieve state-of-the-art performance in yeast. We attribute Ontotype's relatively poor performance on our human data to the sparsity and incompleteness of functional annotations in human; this finding further highlights the strength of our approach where functional relationships are directly inferred from interactomes as opposed to relying on curated annotations. Additionally, we observed that Mashup achieves better overall performance, albeit by a small margin (AUPR of 0.13 compared to 0.1), than Ontotype on the original yeast dataset (Costanzo et al., 2010) used by Yu et al. (2016) (Figure S9).

Notably, our cross-validation was based on the genetic interactions reported by Jerby-Arnon et al. (2014) as ground truth; these were computationally identified based on a diverse set of high-throughput experimental data, including somatic mutations, copy number alterations, gene expression, and short hairpin RNA (shRNA)-based functional screening data. Therefore, most

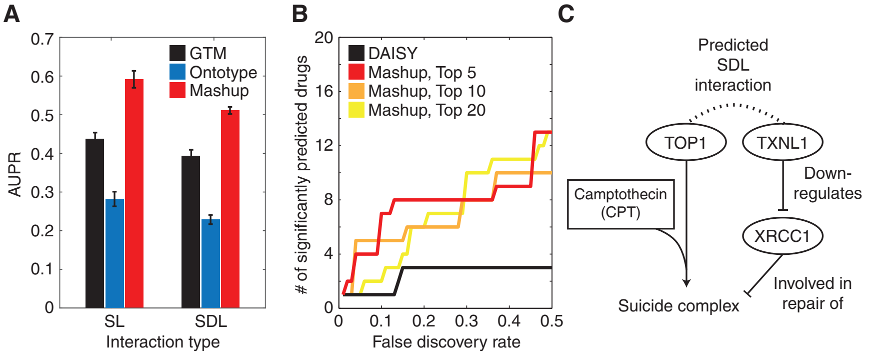


Figure 5. Mashup Improves Genetic Interaction Prediction and Drug Efficacy Prediction

(A) Cross-validation performance of using Mashup representations from STRING networks in SVM classifiers to predict human SL/SDL interactions reported by Jerby-Arnon et al. (2014) as compared to a previous approach that uses various graph-theoretic measures (GTMs) as input features instead (Paladugu et al., 2008) and a more recent approach, Ontotype, that uses the combined, known GO annotations of each gene pair as features in an ensemble of decision trees (Yu et al., 2016). Area under the precision-recall curve (AUPR) is used as the performance metric. Results are summarized over ten trials (SD shown as error bars).

(B) Number of single-target drugs in the Cancer Genome Project data (Garnett et al., 2012) whose efficacy is predicted with statistical significance at varying FDR levels based on SDL interactions originally identified by DAISY (Jerby-Arnon et al., 2014) or top 5, 10, and 20 candidate interactions for each drug target predicted by Mashup-based classifers using DAISY interactions as training data.

(C) An illustration of a putative SDL interaction between TOP1 and TXNL1 predicted by Mashup and its associated literature evidence.

of the "known" interactions in our data were not individually validated in greater depth, which raised a potential concern that Mashup's improvement could be due to statistical artifacts in the high-throughput data. To address this concern, we tested whether Mashup's predicted interactions produce meaningful predictions on an independent biological dataset. In particular, we considered the task of efficacy prediction of cancer drugs; if a drug targets a gene with an SDL interaction, the expression level of the interaction partner is expected to correlate with the efficacy of the drug, which allows us to indirectly validate our predicted SDL interactions by analyzing their ability to predict drug efficacy.

We obtained the efficacy profiles of 50 drugs with single protein targets from the Cancer Genome Project (CGP) (Garnett et al., 2012) over 639 human cancer cell lines. Using an unsupervised prediction framework similar to the one employed by Jerby-Arnon et al. (2014), we calculated how many drug response profiles could be predicted, with statistical significance, using the top SDL interactions predicted by Mashup for each drug (Method Details). We were able to predict many more drugs as compared to using only the interactions identified by DAISY (Jerby-Arnon et al., 2014) (Figure 5B). For instance, at a false discovery rate of 10% and based on the top five candidate interactions, Mashup significantly predicted the efficacy of seven drugs, while DAISY interactions could only predict one. A potential reason for the response of the majority of drugs not being explained in our analysis, as previously noted by Jerby-Arnon et al. (2014), is that many factors other than the essentiality of the drug target, including cell membrane permeability, influence drug efficacy. Note that only 11 out of the 50 single-target drugs had at least one reported SDL interaction. Furthermore, four out of seven drugs whose efficacy we significantly predicted did not have any known SDL interaction partners, which suggests that our classifier was able to produce meaningful predictions even for genes that were not included in the training data. The list of drugs whose response profiles were significantly predicted by Mashup, and their top candidate interactions used for efficacy prediction are provided in Table S2.

TOP1-TXNL1 is one particularly convincing candidate interaction we identified using Mashup that was not significantly identified by other computational methods (Figure 5C). TXNL1, one of the top five candidate interactors for TOP1 (DNA topoisomerase I), has strong evidence in the literature that supports the interaction between the two genes with respect to a drug camptothecin, which targets TOP1. Topoisomerase I normally binds to DNA during transcription to control the topological states of DNA strands. However, in the presence of camptothecin, a normally transient topoisomerase I-DNA complex becomes persistent, resulting in a toxic lesion commonly known as a "suicide complex." It has been noted in the literature that a gene named XRCC1 is involved in the repair of TOP1 suicide complexes (Plo et al., 2003). Furthermore, TXNL1 has recently been observed to downregulate XRCC1 in a gastric cancer cell line (Xu et al., 2014). This finding implies that the higher expression of TXNL1 likely indicates lower levels of XRCC1, which increases the vulnerability of cells to camptothecin-induced DNA damage. Consistent with the literature, the efficacy of camptothecin was significantly predicted in our experiments with TXNL1 as one of the predictors in two independent samples (Table S2). In one of the replicates, the expression level of TXNL1 had strong marginal correlation with the efficacy of camptothecin (Spearman correlation p value = 5.16×10^{-4}). This example illustrates the unique potential of Mashup-based functional inference to produce new biological insights by effectively integrating various types of network data.

DISCUSSION

We have presented Mashup, an integrative framework for analyzing the topology of multiple interaction networks from heterogeneous data sources, which can be used to infer various functional properties of genes or proteins. Mashup characterizes the topology of individual networks by diffusion and then computes compact but highly informative vector representations for nodes in the networks to approximate the diffusion patterns jointly for all networks. We have demonstrated the wide applicability of Mashup in exploiting functional topology in interaction networks by accurately predicting gene function, reconstructing the gene ontology hierarchy, and predicting genetic interactions from heterogeneous network data. We have also demonstrated substantial improvements over previous approaches for each application.

While we have showcased the effectiveness of Mashup as a plug-in architecture for standard tools, such as SVM classifiers and hierarchical clustering, we note that our framework readily allows the use of more sophisticated methods and that such a direction is likely important for further improving performance in many applications. For instance, we recently developed an improved protein function prediction algorithm based on Mashup that exploits the semantic similarity between different functional categories from the ontology hierarchy, which led to significantly better predictions in sparsely annotated GO categories (Wang et al., 2015).

This work was initially inspired by a related line of research in natural language processing. In their seminal paper, Mikolov et al. (2013) introduced a framework that takes a corpus of text documents and gives each word a vector representation based on pairwise co-occurrence patterns and showed that the learned vectors capture semantics of words. In our work, we view genes as words and network diffusion as a way to characterize ‘‘co-occurrence’’ of genes in interaction networks to adapt Mikolov et al.’s idea to real biological networks and demonstrate that the learned features similarly represent functional properties of genes with high accuracy. Mashup generalizes Mikolov et al.’s approach to heterogeneous datasets where the co-occurrence patterns are subdivided into different contexts (different networks). An exciting future direction would be to apply our insights to improve text analysis where different ‘‘types’’ of documents are used to construct more fine-grained co-occurrence data, based on which semantic relationships among words can be more accurately inferred.

For applications in biology, another important direction to explore is incorporating into Mashup other types of information that are not commonly represented as networks, such as sequence, evolutionary, or biochemical properties of individual genes or proteins. Conveniently, this non-network information can be incorporated into our framework in a straightforward manner as additional entries in the feature representation. We also want to emphasize that there is ample opportunity to apply Mashup to other network-based applications, including but not limited to: inter-species network alignment (Liao et al., 2009; Milenković et al., 2010; Singh et al., 2008), protein complex detection (Nepusz et al., 2012), and drug-target interaction prediction (Cheng et al., 2012). Mashup is a versatile tool that provides an effective, unified, and scalable framework for data integration in diverse applications.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- CONTACT FOR REAGENT AND RESOURCE SHARING
- METHOD DETAILS
    - ○ The Mashup Framework
    - ○ Implementation Details
    - ○ Networks and Functional Annotations
    - ○ Gene Function Prediction
    - ○ Gene Ontology Reconstruction
    - ○ Genetic Interaction Prediction
    - ○ Drug Efficacy Prediction
- DATA AND SOFTWARE AVAILABILITY

SUPPLEMENTAL INFORMATION

Supplemental Information includes nine figures, two tables, and one data file and can be found with this article online at http://dx.doi.org/10.1016/j.cels.2016.10.017.

AUTHOR CONTRIBUTIONS

Conceptualization, H.C., B.B., and J.P.; Data Curation, H.C. and J.P.; Investigation, H.C., B.B., and J.P.; Methodology, H.C., B.B., and J.P.; Resources, B.B.; Software, H.C.; Validation, H.C.; Visualization, H.C.; Writing – Original Draft, H.C. and B.B.; Writing – Review & Editing, H.C., B.B., and J.P.; Funding Acquisition, B.B.; Supervision, B.B. and J.P.

ACKNOWLEDGMENTS

A two-page abstract of an earlier version of this work appeared in RECOMB 2015. This work was partially supported by NIH grant R01GM081871.

Received: June 9, 2016
Revised: August 14, 2016
Accepted: October 19, 2016
Published: November 23, 2016

REFERENCES

Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T., et al.; The Gene Ontology Consortium (2000). Gene ontology: Tool for the unification of biology. Nat. Genet. *25*, 25–29.

Beauchamp, M.A. (1965). An improved index of centrality. Behav. Sci. *10*, 161–163.

Benjamini, Y., and Hochberg, Y. (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. J. R. Stat. Soc. B *57*, 289–300.

Berger, B., Peng, J., and Singh, M. (2013). Computational solutions for omics data. Nat. Rev. Genet. *14*, 333–346.

Bonacich, P. (1972). Factoring and weighting approaches to status scores and clique identification. J. Math. Sociol. *2*, 113–120.

Brandes, U. (2001). A faster algorithm for betweenness centrality. J. Math. Sociol. *25*, 163–177.

Brandes, U., and Fleischer, D. (2005). Centrality measures based on current flow. In Proceedings of the 22nd annual conference on Theoretical Aspects of Computer Science, V. Diekert and B. Durand, eds. (Springer), pp. 533–544.

Cao, M., Pietras, C.M., Feng, X., Doroschak, K.J., Schaffner, T., Park, J., Zhang, H., Cowen, L.J., and Hescott, B.J. (2014). New directions for diffusion-based network prediction of protein function: Incorporating pathways with confidence. Bioinformatics *30*, i219–i227.

Chan, D.A., and Giaccia, A.J. (2011). Harnessing synthetic lethal interactions in anticancer drug discovery. Nat. Rev. Drug Discov. *10*, 351–364.

Chang, C., and Lin, C. (2011). LIBSVM: A library for support vector machines. ACM Trans. Intell. Syst. Technol. *2*, 27.

Cheng, F., Liu, C., Jiang, J., Lu, W., Li, W., Liu, G., Zhou, W., Huang, J., and Tang, Y. (2012). Prediction of drug-target interactions and drug repositioning via network-based inference. PLoS Comput. Biol. *8*, e1002503.

Clark, W.T., and Radivojac, P. (2011). Analysis of protein function and its prediction from amino acid sequence. Proteins *79*, 2086–2096.

Costanzo, M., Baryshnikova, A., Bellay, J., Kim, Y., Spear, E.D., Sevier, C.S., Ding, H., Koh, J.L., Toufighi, K., Mostafavi, S., et al. (2010). The genetic landscape of a cell. Science *327*, 425–431.

Davis, J., and Goadrich, M. (2006). The relationship between Precision-Recall and ROC curves. In Proceedings of the 23rd International Conference on Machine Learning, W.W. Cohen and A. Moore, eds. (ICML), pp. 233–240.

Dutkowski, J., Kramer, M., Surma, M.A., Balakrishnan, R., Cherry, J.M., Krogan, N.J., and Ideker, T. (2013). A gene ontology inferred from molecular networks. Nat. Biotechnol. *31*, 38–45.

Enault, F., Suhre, K., and Claverie, J.-M. (2005). Phydbac "Gene Function Predictor": A gene annotation tool based on genomic context analysis. BMC Bioinformatics *6*, 247.

Franceschini, A., Szklarczyk, D., Frankild, S., Kuhn, M., Simonovic, M., Roth, A., Lin, J., Minguez, P., Bork, P., von Mering, C., and Jensen, L.J. (2013). STRING v9.1: Protein-protein interaction networks, with increased coverage and integration. Nucleic Acids Res. *41* (Database issue, D1), D808–D815.

Freeman, L. (1977). A set of measures of centrality based on betweenness. Sociometry *40*, 35–41.

Garnett, M.J., Edelman, E.J., Heidorn, S.J., Greenman, C.D., Dastur, A., Lau, K.W., Greninger, P., Thompson, I.R., Luo, X., Soares, J., et al. (2012). Systematic identification of genomic markers of drug sensitivity in cancer cells. Nature *483*, 570–575.

Gaudet, P., Livstone, M.S., Lewis, S.E., and Thomas, P.D. (2011). Phylogenetic-based propagation of functional annotations within the Gene Ontology consortium. Brief. Bioinform. *12*, 449–462.

Huttenhower, C., Hibbs, M., Myers, C., and Troyanskaya, O.G. (2006). A scalable method for integration and functional analysis of multiple microarray datasets. Bioinformatics *22*, 2890–2897.

Jerby-Arnon, L., Pfetzer, N., Waldman, Y.Y., McGarry, L., James, D., Shanks, E., Seashore-Ludlow, B., Weinstock, A., Geiger, T., Clemons, P.A., et al. (2014). Predicting cancer-specific vulnerability via data-driven detection of synthetic lethality. Cell *158*, 1199–1209.

Kim, H., Shin, J., Kim, E., Kim, H., Hwang, S., Shim, J.E., and Lee, I. (2013). YeastNet v3: A public database of data-specific and integrated functional gene networks for Saccharomyces cerevisiae. Nucleic Acids Res. *42*, D731–D736.

Kramer, M., Dutkowski, J., Yu, M., Bafna, V., and Ideker, T. (2014). Inferring gene ontologies from pairwise similarity data. Bioinformatics *30*, i34–i42.

Lee, I., Blom, U.M., Wang, P.I., Shim, J.E., and Marcotte, E.M. (2011). Prioritizing candidate disease genes by network-based boosting of genome-wide association data. Genome Res. *21*, 1109–1121.

Liao, C.-S., Lu, K., Baym, M., Singh, R., and Berger, B. (2009). IsoRankN: Spectral methods for global alignment of multiple protein networks. Bioinformatics *25*, i253–i258.

Lin, H.-T., Lin, C.-J., and Weng, R.C. (2007). A note on Platt's probabilistic outputs for support vector machines. Mach. Learn. *68*, 267–276.

Macropol, K., Can, T., and Singh, A.K. (2009). RRW: Repeated random walks on genome-scale protein networks for local cluster discovery. BMC Bioinformatics *10*, 283.

Mikolov, T., Sutskever, I., and Chen, K. (2013). Distributed representations of words and phrases and their compositionality. Advances in Neural Information Processing Systems (NIPS). C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, K.Q. Weinberger, eds. Proceedings of the Neural Information Processing Systems, 3111–31119. Retrieved from http://papers.nips.cc/paper/5021-di

Milenković, T., and Pržulj, N. (2008). Uncovering biological network function via graphlet degree signatures. Cancer Inform. *6*, 257–273.

Milenković, T., Ng, W.L., Hayes, W., and Pržulj, N. (2010). Optimal network alignment with graphlet degree vectors. Cancer Inform. *9*, 121–137.

Mostafavi, S., and Morris, Q. (2010). Fast integration of heterogeneous data sources for predicting gene function with limited annotation. Bioinformatics *26*, 1759–1765.

Mostafavi, S., Ray, D., Warde-Farley, D., Grouios, C., and Morris, Q. (2008). GeneMANIA: A real-time multiple association network integration algorithm for predicting gene function. Genome Biol. *9* (*Suppl 1*), S4.

Mostafavi, S., Goldenberg, A., and Morris, Q. (2012). Labeling nodes using three degrees of propagation. PLoS ONE *7*, e51947.

Nepusz, T., Yu, H., and Paccanaro, A. (2012). Detecting overlapping protein complexes in protein-protein interaction networks. Nat. Methods *9*, 471–472.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). The PageRank citation ranking: Bringing order to the Web. Technical Report. Stanford Infolab. Retrieved from http://ilpubs.stanford.edu:8090/422.

Pal, D., and Eisenberg, D. (2005). Inference of protein function from protein structure. Structure *13*, 121–130.

Paladugu, S.R., Zhao, S., Ray, A., and Raval, A. (2008). Mining protein networks for synthetic genetic interactions. BMC Bioinformatics *9*, 426.

Pandey, G., Zhang, B., Chang, A.N., Myers, C.L., Zhu, J., Kumar, V., and Schadt, E.E. (2010). An integrative multi-network and multi-classifier approach to predict genetic interactions. PLoS Comput. Biol. *6*, e1000928.

Park, Y., and Bader, J.S. (2011). Resolving the structure of interactomes with hierarchical agglomerative clustering. BMC Bioinformatics *12* (*Suppl 1*), S44.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourget, V., et al. (2011). Scikit-learn: Machine learning in Python. J. Mach. Learn. Res. *12*, 2825–2830.

Platt, J. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. Advances in Large Margin Classifiers *10*, 61–74.

Plo, I., Liao, Z.Y., Barceló, J.M., Kohlhagen, G., Caldecott, K.W., Weinfeld, M., and Pommier, Y. (2003). Association of XRCC1 and tyrosyl DNA phosphodiesterase (Tdp1) for the repair of topoisomerase I-mediated DNA lesions. DNA Repair (Amst.) *2*, 1087–1100.

Radivojac, P., Clark, W.T., Oron, T.R., Schnoes, A.M., Wittkop, T., Sokolov, A., Graim, K., Funk, C., Verspoor, K., Ben-Hur, A., Pandey, G., et al. (2013). A large-scale evaluation of computational protein function prediction. Nat. Methods *10*, 221–227.

Ruepp, A., Zollner, A., Maier, D., Albermann, K., Hani, J., Mokrejs, M., Tetko, I., Güldener, U., Mannhaupt, G., Münsterkötter, M., and Mewes, H.W. (2004). The FunCat, a functional annotation scheme for systematic classification of proteins from whole genomes. Nucleic Acids Res. *32*, 5539–5545.

Schwikowski, B., Uetz, P., and Fields, S. (2000). A network of protein-protein interactions in yeast. Nat. Biotechnol. *18*, 1257–1261.

Singh, R., Xu, J., and Berger, B. (2008). Global alignment of multiple protein interaction networks with application to functional orthology detection. Proc. Natl. Acad. Sci. USA *105*, 12763–12768.

Sneath, P., and Sokal, R. (1973). Numerical Taxonomy. The Principles and Practice of Numerical Classification (Freeman).

Stephenson, K., and Zelen, M. (1989). Rethinking centrality: Methods and examples. Soc. Networks *11*, 1–37.

Tong, H., Faloutsos, C., and Pan, J. (2006). Fast random walk with restart and its applications. In Proceedings of the Sixth International Conference on Data Mining (IEEE Computer Society), pp. 613–622.

Wang, P.I., Hwang, S., Kincaid, R.P., Sullivan, C.S., Lee, I., and Marcotte, E.M. (2012). RIDDLE: Reflective diffusion and local extension reveal functional associations for unannotated gene sets via proximity in a gene network. Genome Biol. *13*, R125.

Wang, S., Cho, H., Zhai, C., Berger, B., and Peng, J. (2015). Exploiting ontology graph for predicting sparsely annotated gene function. Bioinformatics *31*, i357–i364.

Watts, D.J., and Strogatz, S.H. (1998). Collective dynamics of 'small-world' networks. Nature *393*, 440–442.

Wong, S.L., Zhang, L.V., Tong, A.H.Y., Li, Z., Goldberg, D.S., King, O.D., Lesage, G., Vidal, M., Andrews, B., Bussey, H., Boone, C., et al. (2004). Combining biological networks to predict genetic interactions. Proc. Natl. Acad. Sci. USA *101*, 15682–15687.

Wong, A.K., Krishnan, A., Yao, V., Tadych, A., and Troyanskaya, O.G. (2015). IMP 2.0: A multi-species functional genomics portal for integration, visualization and prediction of protein functions and networks. Nucleic Acids Res. *43* (W1), W128–W133.

Xu, W., Wang, S., Chen, Q., Zhang, Y., Ni, P., Wu, X., Zhang, J., Qiang, F., Li, A., Røe, O.D., et al. (2014). TXNL1-XRCC1 pathway regulates cisplatin-induced cell death and contributes to resistance in human gastric cancer. Cell Death Dis. *5*, e1055.

Yu, D., Kim, M., Xiao, G., and Hwang, T.H. (2013). Review of biological network data and its applications. Genomics Inform. *11*, 200–210.

Yu, M.K., Kramer, M., Dutkowski, J., Srivas, R., Licon, K., Kreisberg, J., Ng, C.T., Krogan, N., Sharan, R., and Ideker, T. (2016). Translation of Genotype to Phenotype by a Hierarchy of Cell Subsystems. Cell Syst. *2*, 77–88.

Zhang, A., Ramanathan, M., Hwang, W., and Cho, Y. (2010). Bridging centrality: A concept and formula to identify bridging nodes in scale-free networks. U.S. patent 7,808,921.

Zhu, C., Byrd, R.H., Lu, P., and Nocedal, J. (1997). Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization. ACM Trans. Math. Softw. *23*, 550–560.

Žitnik, M., and Zupan, B. (2015). Data fusion by matrix factorization. IEEE Trans. Pattern Anal. Mach. Intell. *37*, 41–53.

Žitnik, M., Nam, E.A., Dinh, C., Kuspa, A., Shaulsky, G., and Zupan, B. (2015). Gene prioritization by compressive data fusion and chaining. PLoS Comput. Biol. *11*, e1004552.

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited Data		
STRING database v9.1	STRING Consortium	http://string-db.org
YeastNet v3	Kim et al., 2013	http://www.inetbio.org/yeastnet/
MIPS functional catalog v2.1	Ruepp et al., 2004	http://mips.helmholtz-muenchen.de/funcatDB/
GO hierarchy and gene annotations	Gene Ontology Consortium	http://geneontology.org/page/go-database
NeXO v1.0	Dutkowski et al., 2013	http://www.nexontology.org/
Human SL and SDL interactions	Jerby-Arnon et al., 2014	http://dx.doi.org/10.1016/j.cell.2014.07.027
Yeast genetic interactions	Costanzo et al., 2010	http://dx.doi.org/10.1126/science.1180823
Drug response profiles of cancer cell lines	Garnett et al., 2012	http://dx.doi.org/10.1038/nature11005
Software and Algorithms		
LIBSVM	Chang and Lin, 2011	https://www.csie.ntu.edu.tw/~cjlin/libsvm/
GeneMANIA	Mostafavi and Morris, 2010	http://morrislab.med.utoronto.ca/~sara/SW/
CliXO v3.0	Kramer et al., 2014	https://mhk7.github.io/clixo_0.3/
Ontology alignment algorithm	Kramer et al., 2014	https://mhk7.github.io/alignOntology/
Scikit-learn (random forest classifiers)	Pedregosa et al., 2011	http://scikit-learn.org/stable/
Other		
Code and example data for Mashup	This paper	http://mashup.csail.mit.edu

CONTACT FOR REAGENT AND RESOURCE SHARING

As Lead Contact, Jian Peng is responsible for all reagent and resource requests. Please contact Jian Peng at jianpeng@illinois.edu with requests and inquiries.

METHOD DETAILS

The Mashup Framework

Random Walk with Restart Review

The random walk with restart (RWR) method has been well established for analyzing network structures. By allowing the restart of a random walk from the initial node in each step with a probability, RWR can take into consideration both local and global topology within the network to identify the relevant or important nodes in the network. Let A denote the adjacency matrix of a (weighted) molecular interaction network $G=(V,E)$ with n nodes, each denoting a gene or a protein. Each entry B_{ij} in the transition probability matrix B, which stores the probability of a transition from node j to node i, is computed as

$$B_{ij} = \frac{A_{ij}}{\sum_{i'} A_{i'j}}.$$

Formally, the RWR from a node i is defined as

$$s_i^{t+1} = (1-p_r)Bs_i^t + p_r e_i$$

where p_r is the probability of restart, controlling the relative influence of local and global topological information in the diffusion, with higher chances of restart placing greater emphasis on the local structure; e_i is a n-dimensional distribution vector with $e_i(i)=1$ and $e_i(j)=0$, $\forall j \neq i$; s_i^t is a n-dimensional distribution (column) vector in which each entry holds the probability of a node being visited after t steps in the random walk, starting from node i. The first term in the above update corresponds to following a random edge connected to the current node, while the second term corresponds to restarting from the initial node i. At the fixed point of this iteration we obtain the stationary distribution s_i^∞. Consistent with a previous work (Cao et al., 2014), we define the *diffusion state* $s_i = s_i^\infty \in \Delta_n$ of each node i to be the stationary distribution of RWR starting at each node, where Δ_n denotes the n-dimensional probability simplex. Intuitively, the j th element, s_{ij}, represents the probability of RWR starting at node i ending up at node j in equilibrium. When the diffusion states of two nodes are close to one another, it implies that they are in similar positions within the graph with respect to other nodes, which might suggest functional similarity. This insight provided the basis for several diffusion-based methods (Cao et al., 2014; Macropol et al., 2009; Wang et al., 2012) that aim to predict characteristics of genes or proteins by using the diffusion states to better capture topological associations. Instead of simply using the probability in the diffusion state, the diffusion state distance (DSD) approach, using L1 distances between diffusion states, achieved the state-of-the-art performance on predicting protein functions on yeast interactomes (Cao et al., 2014).

Novel Dimensionality Reduction

A key observation behind our approach is that the diffusion states obtained in this manner are still noisy, in part due to their high dimensionality and the incompleteness of the original network data. With the goal of noise and dimensionality reduction, we approximate each diffusion state s_i with a multinomial logistic model based on a latent vector representation of nodes that uses far fewer dimensions than the original, n-dimensional state. Specifically, we compute the probability assigned to node j in the diffusion state of node i as

$$\hat{s}_{ij} \ := \ \frac{\exp\{x_i^T w_j\}}{\sum_{j'} \exp\{x_i^T w_{j'}\}},$$

where $\forall i, w_i, x_i \in \mathbb{R}^d$ for $d \ll n$. Each node is given two vector representations, w_i and x_i. We refer to w_i as the context feature and x_i as the node feature of node i, both capturing the intrinsic topological properties in the network. If x_i and w_j are close in direction and with large inner product, node j should be frequently visited in the random walk starting from node i. Ideally, if the vector representation w and x is able to capture fine-grain topological properties, we can use it to retrieve genes with similar functions or use it as features for other network-based machine learning applications. While it is possible to enforce equality between these two vectors, decoupling them leads to a more manageable optimization problem and also allows our framework to be readily extended to the multiple network case, which is further described in the next section.

Given this model, we formulate the following optimization problem that takes a set of observed diffusion states $s = \{s_1, \ldots, s_n\}$ as input and finds the low-dimensional vector representation of nodes w and x that best approximates s according to the multinomial logistic model. To obtain w and x for all nodes, we use KL-divergence (relative entropy) as the objective function to minimize, which is a natural choice for comparing probability distributions, to guide the optimization:

$$\underset{w,\,x}{\text{minimize}}\, C(s, \hat{s}) = \frac{1}{n} \sum\nolimits_{i=1}^{n} D_{KL}(s_i \parallel \hat{s}_i).$$

By writing out the definition of relative entropy and $\hat{s}$, we can express the objective as

$$C(s, \hat{s}) = \frac{1}{n} \sum_{i=1}^{n} \left[-H(s_i) - \sum_{j=1}^{n} s_{ij} \left(x_i^T w_j - \log \left(\sum_{j'=1}^{n} \exp\{x_i^T w_{j'}\} \right) \right) \right],$$

where $H(\cdot)$ denotes the entropy.

Novel Integration of Heterogeneous Networks

We can naturally extend our dimensionality reduction framework to integrate network data from diverse sources. We first perform random walks on each individual network k separately and obtain network-specific diffusion states s_i^k for each node i. We also construct the multinomial distribution $\hat{s}_{ij}^k$ from the following logistic model

$$\hat{s}_{ij}^k \ := \ \frac{\exp\left\{x_i^T w_j^k\right\}}{\sum_{j'} \exp\left\{x_i^T w_{j'}^k\right\}},$$

where for each node i in network k, we assign it a network-specific context vector representation w_i^k, which encodes the intrinsic topological properties of network dataset k; for node features x, we allow them to be shared across all K networks. Finally, we jointly optimize the objective function,

$$\underset{w,\,x}{\text{minimize}}\, C(s, \hat{s}) = \frac{1}{n} \sum\nolimits_{k=1}^{K} \sum\nolimits_{i=1}^{n} D_{KL}\left(s_i^k \parallel \hat{s}_i^k\right),$$

and use the optimized node features x for various functional inference tasks. Note that it is possible to weight the divergence term for each network differently, but we give equal importance to each network in this work for simplicity. In addition, while here we assume that all networks are defined over the same set of nodes, given overlapping but different node sets one can take the union of distinct nodes and augment each network with missing nodes to unify the node sets. We believe this approach is preferable to taking the intersection of node sets, as paths over the nodes that are missing in another network could still contain useful topological information to be captured by our diffusion process.

Implementation Details

To optimize the objective function of Mashup, we computed the gradients with respect to the parameters w and x:

$$\nabla_{w_i^k} C(s, \hat{s}) = \frac{1}{n} \sum_{j=1}^{n} \left(\hat{s}_{ji}^k - s_{ji}^k \right) x_j,$$

$$\nabla_{x_i} C(s,\hat{s}) = \frac{1}{n}\sum_{k=1}^{K}\sum_{j=1}^{n}\left(\hat{s}_{ij}^{k} - s_{ij}^{k}\right) w_j^{k}.$$

Both the objective function and the gradients can be computed in $O(n^2dK)$ time. We used a standard quasi-Newton method L-BFGS (Zhu et al., 1997) with these gradients to find the low-dimensional vector representations corresponding to a local optimum of our optimization problem. We used uniform random numbers from $[-0.05,\ 0.05]$ to initialize the vectors and observed that this consistently leads to good solutions.

For the human dataset, we used an alternative objective that allows for more efficient optimization based on singular value decomposition (SVD), in order to cope with large number of genes. We first concatenated the diffusion states for each network k to form a $n\times n$ diffusion state matrix S^k where s_i^k is the i th column. Then, we concatenated the resulting matrices to obtain a $nK\times n$ matrix S. We took the logarithm of each element to obtain $\tilde{S}$ and performed truncated SVD on $\tilde{S}$ (with a user-specified number of components) to get a low-rank factorization $U\Sigma V$. We assigned the columns of $\Sigma^{1/2}U^T$ to $\{w_i^k\}$ and the columns of $\Sigma^{1/2}V$ to $\{x_i\}$. Intuitively, this corresponds to a solution that minimizes the difference between the observed and model distributions as measured by the L2 norm in log space. A small smoothing constant (e.g., reciprocal of the number of genes) was added to each entry in S to avoid taking the log of zero. Further performance optimization can be achieved by calculating the top eigenvectors of $R=\sum_{k=1}^{K}(\tilde{S}^k)^T\tilde{S}^k$, whose eigenvectors correspond to the right singular vectors of $\tilde{S}$ (i.e., $\{x_i\}$). Note $\tilde{S}^k$ denotes the log-transformed matrix of S^k. In order to calculate R, one needs to keep only a single network in memory at a time, which reduces the memory footprint of this approach from $O(n^2K)$ to $O(n^2)$, thereby allowing Mashup to easily scale to a large number of networks. Thus, besides the time it takes to run RWR on each network, which scales linearly with the number of networks and is typically very fast, the running time of calculating Mashup representations via SVD from the diffusion states is constant with respect to the number of networks.

All of our experiments used restart probability of 0.5 for RWR, and unless otherwise noted, we used 500-dimensional vectors for yeast networks and 800-dimensional vectors for human networks, roughly corresponding to 5%–10% of the original number of dimensions. However, we observed that our results are robust to the choice of these parameters (Figures S4 and S5).

Networks and Functional Annotations

We obtained a collection of protein-protein interaction (PPI) networks of yeast and human from the STRING database v9.1 (Franceschini et al., 2013), which is based on a variety of data sources, including high-throughput interaction assays, curated PPI databases, and conserved co-expression. We excluded the network constructed from text mining of academic literature to prevent confounding caused by links based on functional similarity. The resulting collection consisted of six heterogeneous networks over 6,400 genes with the number of edges varying from 1,361 to 314,013 for yeast, and 18,362 genes with the number of edges varying from 3,717 to 1,544,348 for human. Every edge in these networks is associated with a weight between 0 and 1 representing the probability of edge presence, which we factor into the calculation of transition probabilities in the random walk process.

We downloaded functional annotations from Munich Information Center for Protein Sequences (MIPS) (Ruepp et al., 2004) for yeast and the Gene Ontology database (GO; Ashburner et al., 2000) for human. The functional categories in MIPS are organized in a three-layered hierarchy, where the top level (Level 1) consists of the 17 most general functional categories, the second level (Level 2) consists of 74, and the third (Level 3) consists of the 154 most specific categories. We grouped the GO terms for human in a similar fashion to obtain three distinct levels of functional categories of varying specificity, each containing GO terms with 11-30, 31-100, and 101-300 genes, respectively. Within each level, we iteratively removed categories that had Jaccard similarity greater than 0.1 with another category in the same level in order to avoid statistical artifacts arising from overlapping functional categories. Note we propagated annotations over “is a” and “part of” relations in the GO hierarchy for consistency; if a gene is annotated with a GO term, we additionally annotated it with all ancestor terms.

Gene Function Prediction

To predict gene function using the topological feature representations obtained by Mashup, we formulated the task as a multi-label classification problem and applied an off-the-shelf support vector machine (SVM) toolbox, LIBSVM (Chang and Lin, 2011). We trained a binary classifier for each function and obtained per-class probability scores for each gene in the validation set. We used the standard radial basis function (RBF) kernel for the SVMs and performed a nested five-fold cross-validation within the training data to select the optimal parameters via grid search.

The performances of baseline methods—DSD (Cao et al., 2014) and GeneMANIA (Mostafavi and Morris, 2010)—are obtained as follows. For DSD, following the suggestions of the original paper, we obtained the diffusion states using RWR with restart probability of 0.5 and used a L1 distance-based weighted majority voting scheme where the labels assigned to the k most similar genes are combined using the reciprocal of distance between diffusion states as weights to produce per-label confidence scores ($k=10$). Since DSD takes a single network as input, we used STRING’s approach to integrate the networks as a preprocessing step for DSD: we assign $p_{ij}=1-\prod_k(1-p_{ij}^{(k)})$ as the probability of each edge in the combined network, where $p_{ij}^{(k)}$ is the probability associated with the edge (i,j) in network k. For GeneMANIA, we downloaded the MATLAB implementation online (http://morrislab.med.utoronto.ca/~sara/SW) and applied it to our dataset. Since GeneMANIA generates predicted scores for genes that are not directly comparable across different functional labels, we applied the standard Platt calibration (Lin et al., 2007; Platt, 1999), based on the same implementation as the one provided in LIBSVM, to transform the GeneMANIA scores into probability scores before evaluating them in the same manner as other methods.

For each method, we repeatedly held out 20% of the annotated genes as the validation set and used the remaining 80% to predict their functions. We used four different metrics to evaluate the prediction performance: (i) *Accuracy* is measured by assigning top predicted function to each gene in the validation set and measuring how often our prediction is one of the known functions of the gene. (ii) *Micro-averaged F1 score* is calculated by assigning top α predictions to each gene, constructing a 2-by-2 contingency table for each function (treating it as a binary classification task), and computing the F1 score—harmonic mean of precision and recall— on the combined table where each cell is summed across all functions. We used $\alpha = 3$ for the results presented in this paper, following previous work (Cao et al., 2014; Schwikowski et al., 2000). (iii) *Micro-averaged area under the precision-recall curve* (m-PR) is calculated by vectorizing the matrix of predicted confidence scores for each gene-functional category pair and measuring the area under the precision-recall curve constructed based on the resulting vector, which combines the results from all functional categories. (iv) *Macro-averaged area under the precision-recall curve* (M-PR) is calculated by computing the area under the PR curve separately for each function and taking the average across all labels. Since M-PR gives equal weight to all labels, it is less prone (compared to m-PR) to potential biases caused by some functional labels being easier to predict. We chose not to consider the receiver operating characteristic (ROC) curve, because its behavior is closely related to the PR curve and the latter is more appropriate for a classification task with a large skew in the class distribution (Davis and Goadrich, 2006), which is the case for gene function prediction.

Gene Ontology Reconstruction

We reconstructed the gene ontology based on Mashup's feature representations using a standard agglomerative hierarchical clustering algorithm (UPGMA; Sneath and Sokal, 1973), using a cosine distance function which is most appropriate given the use of pairwise inner products in the multinomial logistic model used by Mashup to learn the feature representations. The resulting tree of clusters was pruned (cluster size ≥ 3) and aligned to each type of GO ontology (biological process, molecular function, cellular component) using the ontology alignment algorithm proposed and implemented by Dutkowski et al. (2013). Following their work, only the statistically significant (FDR = 10%) alignments based on a permutation test were used to assess the level of agreement between GO and the reconstructed ontology, which is measured by the harmonic mean of precision (the fraction of reconstructed ontology terms that are aligned) and recall (the fraction of GO terms that are aligned). Overall alignment score was summarized by taking the geometric mean of the alignment scores of the three types of GO ontology.

For fair comparison with NeXO (Dutkowski et al., 2013), we also implemented heuristics to allow multi-way joins and multiple parents in the reconstructed ontology. Given a cluster, we tested whether pairwise distances within a cluster are significantly smaller than those between the cluster and its sibling, using a one-sided rank-sum test. Intermediary nodes in the tree with insignificant *p*-values (> 0.05) were removed, which induces multi-way joins in the ontology. To allow terms to have more than one parent, we adopted a procedure identical to that of Dutkowski et al. (2013), where an additional link is iteratively added between two terms with significant connectivity that are not already on the same path to root.

To obtain the alignment results for NeXO, we downloaded the reconstructed ontology provided on its website (http://www.nexontology.org) and aligned it to GO ontologies using the original authors' alignment algorithm (https://mhk7.github.io/alignOntology). For CliXO, we downloaded the implementation provided by the authors (https://mhk7.github.io/clixo_0.3). Following the recommendations in the original paper (Kramer et al., 2014), we set $\beta = 0.5$ and selected the value of $\alpha \in \{0.001, 0.005, 0.01, 0.015, 0.02, 0.03, 0.05\}$ that resulted in the highest alignment score.

Genetic Interaction Prediction

Following a previous work (Paladugu et al., 2008), we implemented a support vector machine (SVM) classification framework for genetic interaction prediction. Given the Mashup's topological feature representation of each gene, the feature vector for each *pair* of genes is constructed by taking the mean and the absolute difference of the gene features. We trained SVM classifiers with standard radial basis function (RBF) kernel using an off-the-shelf package LIBSVM (Chang and Lin, 2011) with such features as input to distinguish interacting gene pairs from non-interacting ones. For cross-validation, half of the known interactions and a matching number of sampled non-interactions were used to train the classifiers, and their performance was tested on a dataset consisting of the remaining known interactions and a much larger set of non-interactions such that known interactions compose only 5% of the test data. Non-interactions were sampled uniformly at random from the set of gene pairs without known interactions, which relies on the assumption that the number of unobserved interactions is expected to be only a small fraction. We used the area under ROC curve (AUROC) and the area under precision-recall curve (AUPR) on the test set as performance metrics. The variance and cost parameters of SVM were optimized via grid search in a nested cross-validation framework.

As a baseline, we considered using as input topological features a manually curated set of graph-theoretic measures used by Paladugu et al. (2008), which includes: degree (number of direct interactions), clustering coefficient (Watts and Strogatz, 1998), closeness centrality (Beauchamp, 1965), normalized betweenness centrality (Freeman, 1977), eigenvector centrality (Bonacich, 1972), stress centrality (Brandes, 2001), bridging centrality (Zhang et al., 2010), information centrality (Stephenson and Zelen, 1989), and current-flow betweenness centrality (Brandes and Fleischer, 2005). We calculated each measure for each gene based on an integrated PPI network using STRING's Bayesian integration, as previously described. For the baseline model, we also included a few additional features based on shortest distance between genes and the presence of connecting paths of length two (see original paper for more details; Paladugu et al., 2008).

To evaluate the performance of Ontotype (Yu et al., 2016), we first built a hierarchy of GO terms across all three ontologies (biological process, molecular function, and cellular component) using only ‘‘is a’’ and ‘‘part of’’ relationships. Then, we assigned each gene in STRING to its known GO terms and all of their ancestors in the hierarchy. Given a pair of genes, we constructed a feature vector (termed Ontotype) that has 0, 1, or 2 for each GO term representing the number of genes in the pair that are assigned to the term. Following Yu et al. (2016), we used the implementation of random forest classifiers provided by the Python scikit-learn package (Pedregosa et al., 2011) to classify genetic interactions based on the Ontotype features. We explored a wide range of model parameters: {100, 300, 500, 1000} for number of trees, {10, 30, 50, Full} for maximum depth of the trees, and {0.1, 0.3, 0.5} for the fraction of features to consider at each split. To compare with Mashup, we selected the best parameters for SL and SDL interactions, respectively.

Drug Efficacy Prediction

For each drug, we took the target gene’s top 5, 10, 20 SDL interactors predicted by our Mashup-based classifier trained on all SDL interactions identified by Jerby-Arnon et al. (2014) and a matching number of sampled non-interactions. Then, following a similar procedure as Jerby-Arnon et al. (2014), we counted the number of (predicted) interactors that are *overexpressed* in each cell line. The one-sided Spearman correlation p-value between the number of overexpressed interactors and the IC_{50} value of the drug, which reflects drug efficacy, was used as a measure of prediction accuracy. We tried using each of the top five deciles of the expression level observed across all tissues as the threshold for determining overexpression for each gene and selected the most significant among the resulting correlation p values as the final performance score. To assess the statistical significance of our prediction, we sampled the score 10^5 times from a null distribution by using the same number of randomly selected genes as SDL interactors instead. Given the empirical p-values for each of the drugs tested, we used the Benjamini-Hochberg false discovery rate (FDR) controlling procedure (Benjamini and Hochberg, 1995) with varying FDR thresholds to see the number of drugs whose efficacy we could significantly predict given only each tissue’s expression profiles.

DATA AND SOFTWARE AVAILABILITY

A MATLAB implementation of Mashup, pre-trained vectors for various organisms, and a benchmark dataset for function prediction are available for download at: http://mashup.csail.mit.edu and in Data S1.