A Unified Scoring Scheme for Detecting Essential Proteins in Protein Interaction Networks

Hon Nian Chua, Kar Leong Tew, Xiao-Li Li, See-Kiong Ng
Data Mining Department, Institute for Infocomm Research,
21 Heng Mui Keng Terrace, 119613, Singapore
{hnchua,kltew,xlli,skng}@i2r.a-star.edu.sg

Abstract

The essentiality of a gene or protein is important for understanding the minimal requirements for cellular survival and development. Numerous computational methodologies have been proposed to detect essential proteins from large protein-protein interactions (PPI) datasets. However, only a handful of overlapping essential proteins exists between them. This suggests that the methods may be complementary and an integration scheme which exploits the differences should better detect essential proteins. We introduce a novel algorithm, UniScore, which combines predictions produced by existing methods. Experimental results on four Saccharomyces cerevisiae PPI datasets showed that UniScore consistently produced significantly better predictions and substantially outperforming SVM which is one of the most popular and advanced classification technique. In addition, previously hard-to-detect low-connectivity essential proteins have also been identified by UniScore.

1. Introduction

An essential protein (also known as lethal protein) is one that renders the cell unviable upon its removal or lose of functionality. Such proteins provide invaluable insights into the minimal requirements for cellular survival and development. Research experiments [11, 19, 20, 29, 31] have been conducted with respect to the suggestion that essential proteins evolve much slower than other proteins [16, 27], suggesting that they play key roles in the basic functioning of living organisms. Essential proteins are therefore an important class of proteins to study for the defense against human pathogens. In addition, essential proteins (or genes) have also been found to be associated with human disease genes. In a study on human gene morbidity [18], it was found that there is striking similarity between human morbid genes and the essential proteins of *Drosophila melanogaster*. In another study [23], yeast deletion mutants were used to identify 256 new human mitochondrial proteins with a five-fold greater selection than gene expression analysis, showing that it is possible to screen for human disease genes in *Saccharomyces cerevisiae*. A recent study on the dominant and recessive mutants of disease genes [7] also showed that essential proteins tend to have higher correlation with dominant genes.

Identification of essential proteins have significant implications in both basic and translational biological research. However, high-throughput identification of essential proteins has been difficult. Experimental methods for identifying putative essential proteins, such as creating conditional knockouts, are not feasible for large-scale evaluation. As a result, the essentiality profiles for a substantial number of genes are still unknown [13]. A recent cross species study [9] on the protein-protein interaction (PPI) networks of *Saccharomyces cerevisiae*, *Caenorhabditis elegans*, and *Drosophila melanogaster* revealed significant associations between protein evolution, centrality, and its gene essentiality. This suggests the usefulness of PPI networks as a data source for *in silico* detection of essential proteins. In fact, an earlier work [12] had already shown that it is possible to use a connectivity (degree) measure for detecting essential proteins in PPI networks. Subsequently, other computational approaches such as Clustering Coefficient (CC) [30] and Neighborhood Functional Centrality (NFC) [24] have been devised to take advantage of the increasing availability of PPI network data generated by large-scale experiments (such as Yeast Two-Hybrid [6]) to detect essential proteins.

However, we have found that there was only a low overlap between the top ranked essential proteins detected by different methods (Figure 1). On average, there were only 34 overlapping proteins in the top 100 essential proteins predicted by any two methods for the four benchmark PPI datasets.

In fact, the categories of biological functions performed by the essential proteins predicted by different methods were also quite distinct. For instance, the connectivity

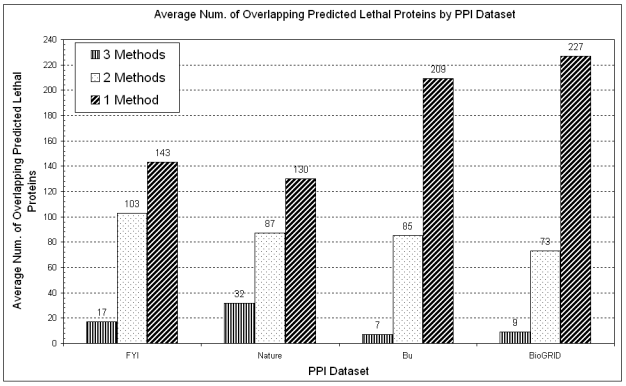

Figure 1. Decreasing number of overlapping proteins in the Top 100 essential proteins predicted by multiple computational methodologies.

measure [12] favored the discovery of lethal proteins with "mitotic cell cycle and cell cycle control" and "fungal and other eukaryotic cell type differentiation" functions. On the other hand, NFC [24] has a preference for lethal proteins with functions that can be broadly grouped as "translation", "replication", and "transcription" categories. Furthermore, existing methods (other than NFC) tend to identify essential proteins with high connectivity in PPI networks. In our core reference list of essential proteins, 46.8% of the essential proteins actually have low connectivity (degree $\leq$ 5). We will discuss on how our UniScore scoring scheme improves the detection of low connectivity essential proteins in Section 4.4.

Given that the current methods detect essential proteins differently, better performance could be obtained by integrating the results from different algorithms [5, 17]. This involves producing a "better" ranking on the combined set of candidates by merging the different rank orderings generated by the different methods. Typically, one assumes that correctly ranked instances will be assigned similar positions in multiple ranking methods. A higher weight is then assigned to rankers that tend to agree with the expert pool, so that the influence of rankers that are less consistent with the rest is reduced [17]. However, this approach will not work in our case as the level of agreement between essential protein ranking methods is low as illustrated in Figure 1. Furthermore, the existing algorithms assign scores with very low granularity, resulting in many proteins with similar scores. The connectivity method, for instance, simply scores proteins with the number of interaction partners. Ranking of such similarly scored proteins is inherently ambiguous. This problem is aggravated by the fact that most proteins have low connectivity (around 60% of the yeast proteins in our datasets are of degrees $\leq$ 5).

To achieve a higher accuracy with the fusion of the results from the multiple detection methods for essential proteins in PPI networks, we need to address the following two difficulties: 1) Each method represents predictions made using information of rather different nature; 2) The scores assigned by each method can differ in both scale and range. In this paper, we propose a novel algorithm UniScore to combine predictions made by different methods for essential protein prediction. Our algorithm takes into account that essential proteins of different functions may be better detected by different methods. By using a common benchmark to reweigh the predictions made by each method as a probability, we devise a probabilistic approach to integrate the predictions made by each individual method. Our experimental results show that UniScore is able to achieve significantly better performance in essential protein detection than any methods being integrated across different datasets.

2. Method

The primary task of the UniScore algorithm is to aggregate the scores of multiple methods for essential protein detection. The input to the UniScore algorithm consists of a set of proteins P, biological functions F and ranking methods R. The UniScore algorithm comprises of three phases. In the first phase, we populate each protein with their Gene Ontology [1] functional annotations such that a protein annotated with a term $f \in F$ will also be annotated with all the ancestor terms of f in the ontology. In the second phase, for each function $f \in F$ and each method $r \in R$, we compute the probability that a protein is essential given that it has a function f and is assigned a certain score by method r. In the third and last phase, we compute a unified score for each protein based on the probabilities computed in the second phase.

2.1. Function Annotation

Functional annotations are usually organized in hierarchical structures such as a Direct Acyclic Graph (GO) or a Tree (MIPS). Annotations in hierarchical structures follow the "true path" rule [4] – a protein annotated with a functional term is also annotated with all its ancestor terms. While this is implied implicitly, annotations from databases usually only provide the most specific annotations. In the first phase of the UniScore algorithm, we make sure that all functional annotations are appropriately propagated.

Definition 1 *For each protein $p \in P$, let F_p represents the set of functions annotated to p where $F_p \subseteq F$.*

Definition 2 *For each F_p, we compute its superset F'_p by including the ancestors of each function $F \in F_p$ into F_p.*

$$F'_p = F_p \cup \bigcup_{f \in F_p} A_f$$

where A_f is the set of ancestors of function f.

2.2. Unified Weighting Scheme

Each input method assigns a score to each protein during prediction. However, as mentioned previously, the reliability of each method, as well as the range and scale of the scores they assign to proteins can differ. Furthermore, as each method uses a different kind of information for predicting the essentiality of a protein, they may work better for proteins with certain biological functions.

In this phase, we estimate the likelihood that protein *p* is essential given the observation that a method *r* assigns a score *s* to *p* with function *f*. To do this, we can examine all proteins that are assigned scores *s* by method *r*, and have function *f* in the training data and compute the fraction of these proteins that are essential. However, the scores assigned by each prediction method may not be discrete, and there may not be many proteins that are assigned the same score. Thus, it is necessary that we first group proteins with similar scores together.

2.2.1 Grouping proteins with similar scores

Proteins with function *f* that are assigned similar scores by a ranking method *r* are grouped into groups of at least size μ in the following way:

1. Sort the set of proteins P_f with function *f*, based on the ranking score
2. From the list of sorted proteins, $p_0, ..., p_n$, we create a new group G_0^{rf}, and insert $p_0, ..., p_{u-1}$ into G_0^r (first insert μ members into a initial group). Continue to insert the proteins $P_{\mu+k}$, $0 \leq k < n - \mu - k$, into G_0^{rf} as long as $S_k(p_{\mu+k}) = S_k(p_{\mu-1})$ (then add new members if their scores are equal to the last member in the initial group).
3. Starting from $P_{\mu+k}$, create a new group G_1^{rf} and repeat step 2.
4. If the last group G_m^{rf} has a size smaller than μ, it is merged with G_{m-1}^{rf}.

The target here is to provide the same value for each entity found within the same group as the difference between their scores are small. This will fulfill our initial intention of discretizing the scores.

2.2.2 Computing the confidence of each protein group

The likelihood that a protein *p* is essential given that it is annotated with function *f* and is assigned score $S_r(p)$ by ranking method *r*, is then computed by:

$$P_p(essential|f, S_r(p)) = \frac{\sum_{p \in G_k^{rf}} Essential(p)}{|G_k^{rf}| + 1} \quad (1)$$

where $Essential(p)$ returns 1 if *p* is an essential protein, 0 otherwise; G_k^{rk} is the group of proteins with function *f* in which $S_r(p)$ belongs to. It computes the fraction of the proteins in each group that are essential.

We estimate the likelihood that a protein *p* is essential given that it is assigned score $S_r(p)$ by ranking method *r*, as:

$$P_p(essential|S_r(p)) = \max_{f \in F'_p}(P_p(essential|f, S_r(p))) \quad (2)$$

This is the maximum value from equation (1) among all functions of *p*.

2.3. Integrating the confidences of multiple methods

Finally, we compute the likelihood that a protein *p* is essential given that it is assigned score $S_n(p)$ by multiple methods $r_i (i = 1, ..., n)$ using a Bayesian based approach:

$$P_p(essential) = 1 - \prod_{r \in R}(1 - P_p(essential|S_r(p))) \quad (3)$$

A higher support value will indicate a higher probability that protein *p* is essential.

3. Experimental Data

To evaluate the performance of our proposed UniScore algorithm, we perform predictions on the essentiality of proteins from *Saccharomyces cerevisiae* (Bakers' Yeast), as it has been well characterized by knockout experiments (from which our core reference list of essential proteins was derived). *S. cerevisiae* has also been widely used in existing works on computational inference and evaluations of essential proteins [9, 12, 13, 23, 30].

3.1. Datasets

Protein-protein interactions: We used four publicly available *Saccharomyces cerevisiae* PPI datasets for evaluation and name them according to the method/source that

they were obtained from: Filtered Yeast Interactions — *FYI* [10], *Nature* [12], *Bu* [3], and *BioGRID* [22]. Details on each dataset are presented in Table 1.

The first dataset *FYI* is a high-quality (reliable) but sparse yeast interaction dataset with minimal false positive interactions [10]. The *Nature* dataset is also spare; we have included it in our experiments as it was used by the previous work on validating the connectivity measure [12] for essential protein prediction. The third dataset *Bu*, originally compiled for function prediction [3], is a relatively larger network with 3 times as many interactions as the previous two datasets. The fourth dataset BioGRID (version 2.0.33) was downloaded from BioGRID [22], giving a more recent PPI dataset derived from various biological experiments. We pre-processed the datasets by removing self-interacting interactions and isolated protein pairs. For the functional annotation of the proteins, we used functions classified as biological process by GO (submitted 29/03/2008) [1].

Table 1. Details of the four protein interaction datasets used in our evaluation experiments.

	# Proteins	# Essential	# Interactions
FYI	1210 (*958*)	464 (*333*)	2400
Nature	1638 (*1490*)	369 (*312*)	2201
Bu	2224 (*1531*)	670 (*349*)	6609
BioGRID	4914 (*2108*)	992 (*174*)	37826

Note: Italicized numbers in brackets represents proteins with connectivity ≤ 5.

Essential Proteins: For evaluation, we used a benchmark list of $1,106$ known essential proteins for *Saccharomyces cerevisiae* [8]. This set of essential proteins was derived experimentally using PCR-based gene deletion strategy [2, 26]. We will refer to this list as the core reference list of essential proteins.

3.2. Existing Ranking Methods

To show the power of integrating multiple prediction methods, we used four previously published essential protein detection method, and one unpublished method to train our model. The methods are Degree [12], Clustering Coefficient (CC) [30], Neighborhood Functional Centrality (NFC) [24], and Functional Diversity (FD).

The Degree method was based on the fact that the essentiality of a protein is positively correlated to its connectivity (or degree) in the protein interaction network [12]. The Clustering Coefficient method quantifies the probability of two interacting proteins that are also interacting with a similar third protein [30]. The Neighborhood Functional Centrality is a measure which takes into consideration the functional role a protein plays in terms of its surrounding neighborhood [24]. The Functional Diversity method is based on the assumption of essential proteins performing the role of multi-functional components within a protein interaction network:

Definition 3 *For each protein* p*, FD is computed as follows:*

$$FD(p) = \frac{1}{\sum_{i=1} |F_p| \sum_{j=i+1}} |F_p| RSS(f_i, f_j) \quad (4)$$

where $RSS(f_i, f_j)$ *returns the relative relative specificity similarity score between two functions* f_i *and* f_j [28].

The FD method was not published due to poorer performance when compared to NFC. However, we observed that the FD method can detect different essential proteins. We therefore include it in this experiment to show that our UniScore algorithm can take advantage of multiple prediction methods even if some of the individual methods are not amongst the best.

4. Results

For evaluation, we first compare our UniScore algorithm against the individual ranking methods. We employ the following standard statistical measures: Receiver Operating Characteristic (ROC) curves and the corresponding Areas under the ROC Curve (AUC) values, and Recall vs. Precision curves, to evaluate the performance of each method (Section 4.1). Next, we investigate the effect of group size on the performance of UniScore (Section 4.3). Lastly, we investigate the sensitivity of each method in detecting low-connectivity essential proteins, and how UniScore can significantly improve the performance (Section 4.4).

For our experiments, we used a ten-fold cross-validation to train and determine the weight for each ranking method. The results presented are obtained with the size of each group μ set to be 0.5% of the entire protein population in each dataset. Implications with regard to the change in performance due to group size are covered in Section 4.3.

As UniScore is an integrative method, we choose to compare its performance against Support Vector Machines (SVM) [25] where the score of each method is treated as a feature. We use the default parameters of SVM^{perf} [15] and the results shows that UniScore is much better than SVM in essential proteins detection (Section 4.2).

4.1. Prediction Performance

UniScore is used to combine the predictions from four ranking methods, namely, Connectivity, Clustering Coefficient (CC), Neighborhood Functional Centrality (NFC), and Functional Diversity (FD). The prediction performances of

UniScore, as well as the individual ranking methods are then evaluated against the core reference list of essential proteins.

Table 2. AUC for predictions made using UniScore, Connectivity, SC, CC, NFC, and FD

	FYI	Nature	Bu	BioGRID
Connectivity	60.6%	61.0%	66.0%	72.8%
CC	55.8%	58.6%	58.7%	67.0%
NFC	70.4%	74.7%	77.4%	74.8%
FD	55.8%	59.7%	61.3%	60.9%
SVM	71.0%	74.8%	78.2%	76.4%
UniScore	81.5%	82.8%	82.5%	82.8%

As shown in Table 2, UniScore is on average 8.1% higher than NFC in terms of AUC value, the best of the individual methods. The results clearly demonstrate that integrating the multiple ranking strategies with UniScore can improve the predictions significantly.

Figure 2 presents the ROC curve for each computational method on all four experimental datasets. Through the graph, we can see a clear gap in terms of performance between UniScore and all other methods. A similar performance is also illustrated in Figure 3 where UniScore once again demonstrated its proficiency at detecting essential proteins. We can clearly see that UniScore has consistently excelled in lethal protein predictions across datasets of varying size and quality.

4.2. Comparison with SVM

Each computational method produces a value which describes a specific characteristic of the interaction neighborhood of the protein. As such, each computational value can be regarded as a feature associated with the protein. This features can be used as input to various classifiers such as Support Vector Machines (SVM) [25], Decision Trees, Neural Networks, etc. In this paper, we compare UniScore against SVM which has proven to be one of the best classifiers in many application domains.

Each of the four computational values (Connectivity, CC, NFC, FD) are assembled into a feature vector for each protein and used as an input to SVM. For our experiments, we used SVM^{perf} [15] (version 2.10), which is the high performance version of the popular SVM^{light} [14] implementation. Through empirical experimentations, we found the best performance by setting the parameters $c = 20.0$ (trade-off between training error and margin), and $l = 10$ (optimize the results for ROC measure which is our basis for comparison). The corresponding results are included in Table 2. SVM is able to achieve a higher prediction as compared to each individual method (mixed results when compared against NFC), UniScore is still able to achieve a higher accuracy of 10.5%, 8.0%, 4.3%, and 6.4% for FYI, Nature, Bu, and BioGRID respectively. Which indicates our technique can be effectively used to predict the essential proteins.

4.3. Group Size

Recall that UniScore uses a parameter μ to determine the size of each individual group (Section 2.2.1). Generally, a smaller μ provides higher granularity during the weight estimation, but runs a risk of getting a less accurate estimate for the confidence of each group, and vice versa. To study the effect of μ on the performance of UniScore, we evaluate the performance of UniScore on the four protein-protein interaction datasets with different μ values ranging from 0.25% to 5% of the total number of proteins. The corresponding AUC scores are presented in Table 3. Among the μ values that we have experimented on, $\mu = 0.5$ and $\mu = 0.25$ results in the best prediction performance across all four datasets. The mean deviation of the resulting AUC values is between 1.29% to 1.84%. We can thus see that while the size chosen for μ has some effect on the AUC scores of the predictions made by UniScore, it is not crucial to the performance of UniScore.

Table 3. AUC for predictions made using UniScore with a different μ (as a percentage of the number of proteins) values on four datasets

	FYI	Nature	Bu	BioGRID
μ=5	76.0%	78.7%	78.7%	78.4%
μ=4	78.9%	79.3%	78.8%	78.4%
μ=3	78.5%	80.3%	78.7%	78.9%
μ=2	79.1%	80.8%	79.5%	80.6%
μ=1	81.8%	82.1%	81.8%	79.8%
μ=0.5	81.5%	82.8%	82.5%	82.8%
μ=0.25	81.5%	82.3%	83.2%	84.5%
Mean Deviation	1.70%	1.29%	1.75%	1.84%

4.4. Low connectivity proteins

As we can see from Table 1, a significantly large proportion (61.0% on average) of proteins in the PPI datasets are of low-connectivity (i.e. number of interaction partners ≤ 5). An average of 46.8% of the essential proteins in our core reference list also has low connectivity in the underlying PPI networks, even in denser PPI networks such as Bu and BioGRID. Essential proteins with low-connectivity are special in that although their involvement in the network is

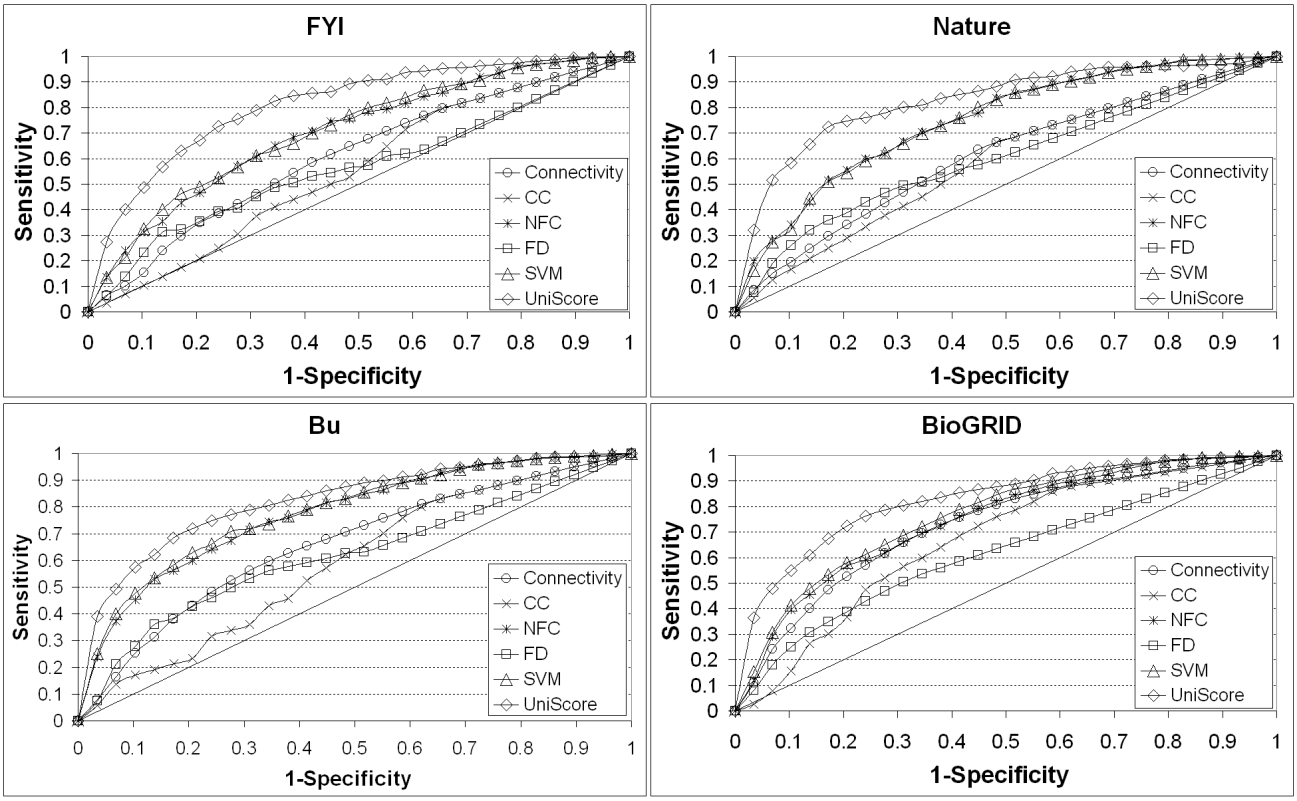

Figure 2. ROC curves for predictions made using UniScore, Connectivity, CC, NFC, and FD on the four different datasets.

low, their presence and functionality is vital to cellular survival and development. This also implies their potential as drug targets as they have bigger impact with less disruption to the underlying PPI network. Low-connectivity proteins are largely missed out by current connectivity-based detection method except the NFC which was shown to better detect the low-connectivity essential proteins than the existing methods [24].

Table 4. Number of low-connectivity essential proteins (degree $\leq$ 5) detected by UniScore and NFC (italicized numbers in brackets) for each dataset in the top N ranked proteins

	FYI	Nature	Bu	BioGRID
50	32 (*4*)	29 (*11*)	22 (*2*	1 (*0*)
100	61 (*16*)	59 (*33*)	37 (*3*)	1 (*0*)
150	89 (*31*)	87 (*50*)	52 (*10*)	2 (*2*)
200	105 (*45*)	107 (*67*)	67 (*19*)	4 (*2*)

Table 4 shows that a further improvement in the prediction of low-connectivity essential proteins can be achieved with UniScore. In fact, UniScore was able to detect some of those essential proteins with connectivity of 1 which had been near impossible to be detected by all the existing methods. Through the fusion of various scores with UniScore, we now have the following numbers of putative connectivity-1 essential proteins detected: *FYI* (9), *Nature* (12), *Bu* (5) in the top 100 predicted essential proteins. There were no connectivity-1 essential proteins detected in BioGRID – this could be due to the relatively small number (174 of low connectivity essential proteins in that particular dataset as shown in Table 1).

5. Discussion & Conclusions

We introduced UniScore, a probabilistic method that integrates the predictions from multiple essential protein prediction methods to achieve better predictions. Results from the extensive experiments showed that UniScore performs much better than the individual methods, confirming that integrating different ranking strategies can indeed provide better results. In addition, we also showed that UniScore was able to predict low-connectivity essential proteins much better than existing methods. Low-connectivity essential proteins may be of special interest as drug targets as they can potentially bring about bigger impact with minimal side-effects.

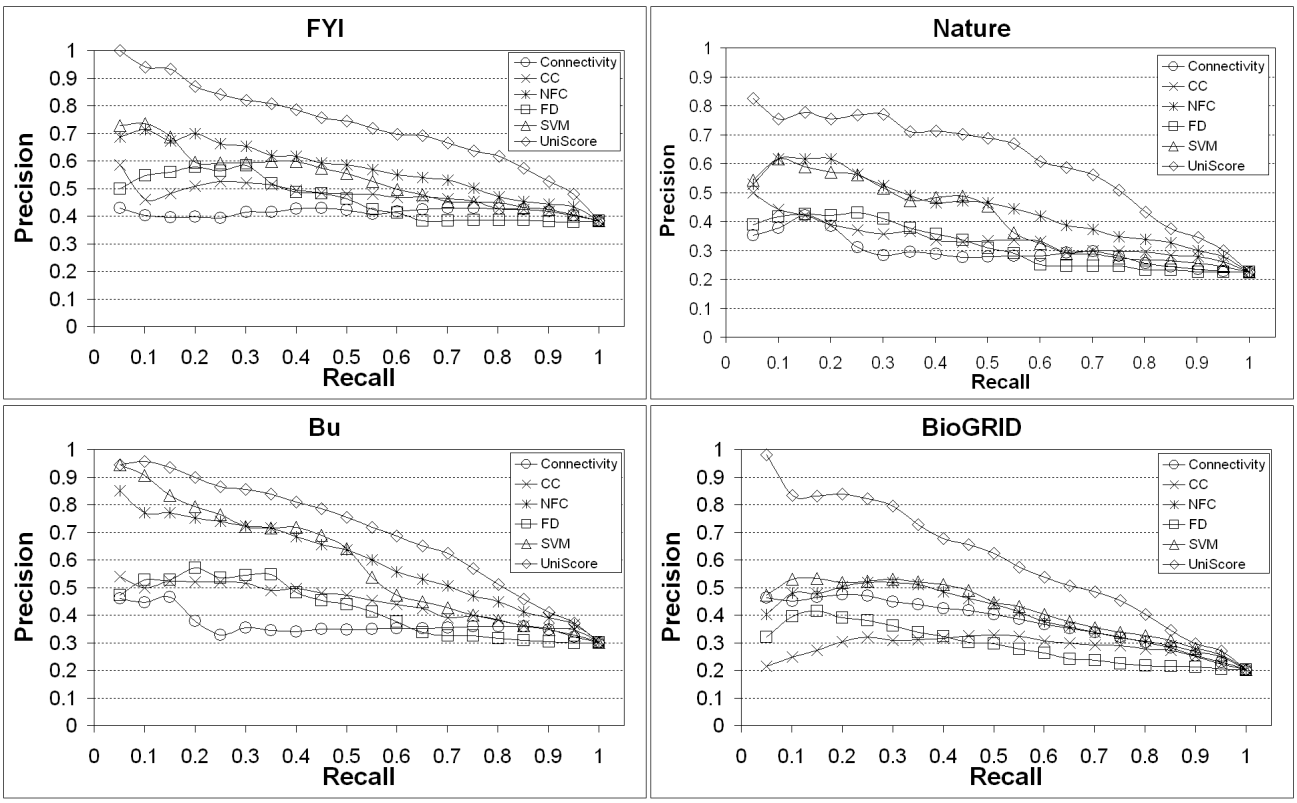

Figure 3. Precision vs Recall curves for predictions made using UniScore, Connectivity, CC, NFC, and FD on the four different datasets.

Although we have successfully integrated the scores of each prediction method, we did not completely resolve limitations of each method. The characteristics of proteins that are highly ranked with UniScore is dependent on the set of prediction methods chosen. A good selection on the set of methods to be used in UniScore will help ensure proteins are scored based on relevant characteristics.

Given that the essentiality profiles of yeast genes are not complete [13], many of the false positives in the high ranking essential proteins predicted by UniScore may be novel essential proteins. Some of these proteins could also be part of a larger essential component (e.g. a complex) that requires multiple deletion/mutation to result in lethality. To test this hypothesis, we performed a preliminary investigation on the relation between the UniScore value of a protein and its likelihood to be associated with synthetic lethality. We obtained the list of synthetic lethality interactions from BioGRID (version 2.0.33). We then grouped proteins based on discretized UniScore values from 0 to 0.9 in steps of 0.1 and computed the fraction of proteins in each group that were associated with synthetic lethality. The results are displayed in Figure 4. We observed that the highly scored proteins are more likely to be involved in synthetic lethality, which provided some affirmation on our earlier suspicion. This suggests that our predictions can also identify to a certain extend proteins that are part of an essential pair of proteins.

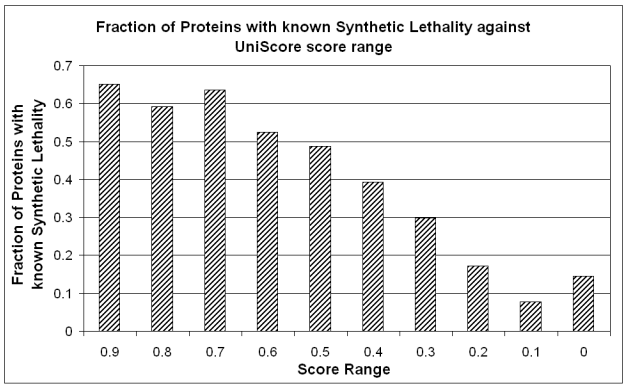

Figure 4. Distribution of the number of proteins identified by known synthetic lethality screens against discretized ranges of UniScore scores.

In this work, we have focused on essential protein prediction methods using the PPI data. It is possible that

other types of biological data can also be exploited to detect essential proteins. For instance, comparative genomics studies have revealed conservation of essential proteins across species such as *Saccharomyces cerevisiae* and *Saccharomyces mikatae* [21], suggesting evolution information can be useful in the detection of essential proteins.

As future work, it would be interesting to see how this information, as well as other prediction methodologies from heterogenous data sources, can be integrated with UniScore for the prediction of essential proteins.

6. Acknowledgement

We like to thank our colleague Sintiani Dewi Teddy for proof reading the manuscript.

References

[1] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, et al. Gene Ontology: tool for the unification of biology. *Nature Genet*, 25:25–29, May 2000.

[2] A. Baudin, O. Ozier-Kalogeropoulos, A. Denouel, F. Lacroute, and C. Cullin. A simple and efficient method for direct gene deletion in Saccharomyces cerevisiae. *Nucl. Acids Res.*, 21(14):3329–3330, July 1993.

[3] D. Bu, Y. Zhao, L. Cai, H. Xue, X. Zhu, H. Lu, J. Zhang, S. Sun, L. Ling, N. Zhang, G. Li, and R. Chen. Topological structure analysis of the protein-protein interaction network in budding yeast. *Nucl. Acids Res.*, 31(9):2443–2450, May 2003.

[4] T. G. O. Consortium. Creating the Gene Ontology Resource: Design and Implementation. *Genome Research*, 11:1425–1433, 2001.

[5] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar. Rank Aggregation Revisited. *Manuscript*, 2001.

[6] S. Fields and S. Ok-Kyu. A novel genetic system to detect protein-protein interactions. *Nature*, 340:245–246, 1989.

[7] S. J. Furney, M. M. Alba, and N. Lopez-Bigas. Differences in the evolutionary history of disease genes affected by dominant or recessive mutations. *BMC Genomics*, 7(165), July 2006.

[8] G. Giaever, A. M. Chu, L. Ni, C. Connelly, et al. Functional profiling of the Saccharomyces cerevisiae genome. *Nature*, 418(6896):387–391, Jul 2002.

[9] M. W. Hahn and A. D. Kern. Comparative Genomics of Centrality and Essentiality in Three Eukaryotic Protein-Interaction Networks. *Mol. Biol. Evol.*, 22(4):803–806, Dec 2004.

[10] J.-D. J. Han, N. Bertin, T. Hao, D. S. Goldberg, G. F. Berriz, L. V. Zhang, D. Dupuy, A. J. M. Walhout, M. E. Cusick, F. P. Roth, and M. Vidal. Evidence for dynamically organized modularity in the yeast protein-protein interaction network. *Nature*, 430(6995):88–93, Jul 2004.

[11] L. D. Hurst and N. G. Smith. Do essential genes evolve slowly? *Curr Biol*, 9:747–750, July 1999.

[12] H. Jeong, S. P. Mason, A. L. Barabasi, and Z. N. Oltvai. Lethality and centrality in protein networks. *Nature*, 411(6833):41–42, May 2001.

[13] H. Jeong, Z. Oltvai, and A.-L. Barabsi. Prediction of Protein Essentiality Based on Genomic Data. *Complexus*, 1(12):19–28, 2003.

[14] T. Joachims. Making large-scale support vector machine learning practical. *Advances in Kernel Methods: Support Vector Machines*, 1998.

[15] T. Joachims. A Support Vector Method for Multivariate Performance Measures. *Proceedings of the International Conference on Machine Learning (ICML)*, 2005.

[16] M. Kimura and T. Ota. On some principles governing molecular evolution. *Proc Natl Acad Sci USA*, 71(7):2848–2852, July 1974.

[17] A. Klementiev, D. Roth, and K. Small. An Unsupervised Learning Algorithm for Rank Aggregation. *Proceedings of Machine Learning: ECML*, pages 616–623, 2007.

[18] F. A. Kondrashov, A. Y. Ogurtsov, and A. S. Kondrashov. Bioinformatical assay of human gene morbidity. *Nucl. Acids Res.*, 32(5):1731–1737, Mar 2004.

[19] B. Y. Liao, N. M. Scott, and J. Z. Zhang. Impacts of gene essentiality, expression pattern, and gene compactness on the evolutionary rate of mammalian proteins. *Mol. Biol. Evol.*, 23(11):2072–2080, 2006.

[20] C. Pal, B. Papp, and L. D. Hurst. Genomic function:Rate of evolution and gene dispensability. *Nature*, 411(6841):1046–1049, Jan 2003.

[21] M. Seringhaus, A. Paccanaro, A. Borneman, M. Snyder, and M. Gerstein. Predicting essential genes in fungal genomes. *Genome Research*, 16:1126–1135, August 2006.

[22] C. Stark, B. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. Tyers. BioGRID: a general repository for interaction datasets. *Nucl. Acids Res.*, 34:D535–D539, 2006.

[23] L. M. Steinmetz, C. Scharfe, A. M. Deutschbauer, D. Mokranjac, Z. S. Herman, et al. Systematic screen for human disease genes in yeast. *Nature Gene*, 31:400–404, Aug 2002.

[24] K. L. Tew, X.-L. Li, and S.-H. Tan. Functional centrality: Detecting lethality of proteins in protein interaction networks. *Genome Informatics*, 19:166–178, 2007.

[25] V. N. Vapnik. The Nature of Statistical Learning Theory. *Springer*, 1995.

[26] A. Wach, A. Brachat, R. Pohlmann, and P. Philippsen. New heterologous modules for classical or PCR-based gene disruptions in Saccharomyces cerevisiae. *Yeast*, 10:1793–1808, 1994.

[27] A. C. Wilson, S. S. Carlson, and T. J. White. Biochemical evolution. *Annu Rev Biochem*, 46:573–639, 1977.

[28] X. Wu, L. Zhu, J. Guo, D. Y. Zhang, and K. Lin. Prediction of yeast protein-protein interaction network: insights from the Gene Ontology and annotations. *Nucl. Acids Res.*, 34:2137–2150, 2006.

[29] J. Yang, Z. L. Gu, and W. H. Li. Rate of protein evolution versus fitness effect of gene deletion. *Mol. Biol. Evol.*, 25(5):772–774, 2003.

[30] H. Yu, D. Greenbaum, H. Xin Lu, X. Zhu, and M. Gerstein. Genomic analysis of essentiality within protein networks. *Trends Genet*, 20(6):227–231, 2004.

[31] J. Z. Zhang and X. L. He. Significant impact of protein dispensability on the instantaneous rate of protein evolution. *Mol. Biol. Evol.*, 22(4):1147–1155, 2005.