Published online 28 May 2008 *Nucleic Acids Research, 2008, Vol. 36, Web Server issue* ***W411–W415***
doi:10.1093/nar/gkn281

PIE: an online prediction system for protein–protein interactions from text

Sun Kim[1], Soo-Yong Shin[2], In-Hee Lee[1], Soo-Jin Kim[3], Ram Sriram[2] and Byoung-Tak Zhang[1,3,*]

[1]Biointelligence Laboratory, School of Computer Science and Engineering, Seoul National University, Seoul 151-744, Korea, [2]Manufacturing Systems Integration Division, National Institute of Standards and Technology, Gaithersburg, MD 20899, USA and [3]Center for Bioinformation Technology, Graduate Program in Bioinformatics, Seoul National University, Seoul 151-742, Korea

Received February 07, 2008; Revised April 16, 2008; Accepted April 26, 2008

ABSTRACT

Protein–protein interaction (PPI) extraction has been an important research topic in bio-text mining area, since the PPI information is critical for understanding biological processes. However, there are very few open systems available on the Web and most of the systems focus on keyword searching based on predefined PPIs. PIE (Protein Interaction information Extraction system) is a configurable Web service to extract PPIs from literature, including user-provided papers as well as PubMed articles. After providing abstracts or papers, the prediction results are displayed in an easily readable form with essential, yet compact features. The PIE interface supports more features such as PDF file extraction, PubMed search tool and network communication, which are useful for biologists and bio-system developers. The PIE system utilizes natural language processing techniques and machine learning methodologies to predict PPI sentences, which results in high precision performance for Web users. PIE is freely available at `http://bi.snu.ac.kr/pie/`.

INTRODUCTION

Protein–protein interaction (PPI) information is critical for understanding the function of individual proteins and the organization of entire biological processes. A large amount of biomedical literature describes PPI experiments, and the protein interaction databases such as IntAct and MINT have been developed by utilizing these biomedical articles. However, the rapid growth of the literature makes it difficult to manually find the necessary information (1). In addition, the dynamic nature of biology makes the ontology or the database building more difficult. With the implementation of automatic analysis initiatives, the amount of information in terms of biological data availability is overwhelming, as reflected by hundreds of databases and Web servers (2). However, despite of the importance of the PPI extraction task, only a few systems are freely available on the Web (3).

Most of existing PPI systems can be divided into two categories: co-occurrence-based approaches and rule-based approaches (4–6). Co-occurrence approaches assume that co-occurrence of gene/protein names in text corresponds to a biological relationship. Rule-based approaches utilize predefined phrase pattern rules. However, these approaches can only extract well-known patterns but may not be able to find new emerging PPIs.

Recently, it has been shown that PPI information has its own pattern at the article and sentence levels (7). Machine learning (ML) techniques are useful for discovering the hidden patterns from training data. ML techniques also provide robust results for unknown patterns. In this article, we describe an online Web service–PIE (Protein interaction information extraction system)–for providing biologists with extracted PPI sentences from text. Our system combines both co-occurrence approaches and rule-based approaches in an ML framework. Co-occurrence models are used for calculating similarities among texts in boosting and support vector machines (SVMs). Rule-based approaches are used in tree kernels to support natural language processing properties. Besides, PIE can automatically find the hidden patterns without predefined rules or patterns by using ML techniques. As a result, PIE performs high precision predictions, which is especially required for Web-based retrieval systems.

From the online service perspectives, PIE contains several novel features. While previous PPI services are mostly based on keyword-based searching on predefined PPIs, PIE does not use locally saved PPIs, but rather

*To whom correspondence should be addressed. Tel: +82 2 880 1847; Fax: +82 2 875 2240; Email: btzhang@bi.snu.ac.kr

The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors

focuses on PPI sentence predictions from the biomedical literature such as user-provided papers and PubMed articles. This feature provides much flexibility for the biologists who are interested in summarizing unknown PPI information out of papers or abstracts. In addition, PIE implements keyword-based extraction, which is similar to the one in other PPI services. By accepting keywords from users, PIE retrieves PubMed database on-the-fly and processes all or part of articles to predict PPI sentences. After uploading abstracts or papers, the prediction results are displayed by highlighting potential PPI sentences. The PIE interface is carefully designed to help biologists and bio-system developers, featuring PDF and HTML support, customized PubMed searching, PPI visualization and network communication.

METHODS

Figure 1 shows a schematic overview of PIE. Two core modules of the system are the article filter and the sentence filter, which predict whether given articles or sentences contain PPI information. The search engine in PIE is implemented to retrieve the stored information such as learning data (article DB) and protein names (protein DB). The Web interface module manages the whole process of PPI predictions from Web users. A part of prediction results is linked to the iHOP service (`http://www.ihop-net.org`) (8). For the PubMed article searching, PIE connects to the online PubMed service using NCBI's E-Utilities (`http://www.ncbi.nlm.nih.gov/entrez/query/static/eutils_help.html`), which will be eventually changed to local PubMed searching for reducing internet traffic loads. The XML–RPC module is responsible for communicating with other PPI services using remote procedure calls (RPCs).

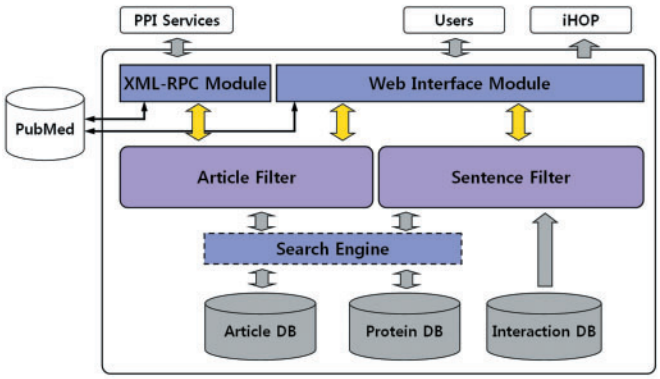

Figure 1. Overview of PIE. The PIE system consists of several modules. 'Article Filter' and 'Sentence Filter' decide whether given articles or sentences contain PPI information. 'Search Engine' retrieves the stored information such as learning data (Article DB) and protein names (Protein DB). 'Interaction DB' means the database including interaction-related words. 'XML–RPC Module' is responsible for RPC communication with other PPI services. 'Web Interface Module' manages the whole process of PPI predictions from Web users. Prediction results contain the links to the iHOP service to provide further protein information. For PubMed search, PIE retrieves PubMed articles using the NCBI E-Utilities.

PIE uses the article filter to increase filtering speed and to enhance system efficiency because the sentence filter is computationally intensive. Brief procedures of the article and sentence filters are presented in the following subsections.

Article filter

In the first step, the article filter classifies whether a given text contains PPI-related information. In doing so, it should not miss any PPI relevant documents, even though a certain amount of irrelevant documents is included. To handle this tradeoff, our system utilizes a cost-sensitive learning algorithm–AdaCost (9)–which provides the flexibility between precision and recall rates. The naive Bayes method (10) is adopted as a weak learner, which is known to be efficient in text filtering. The ensemble of naive Bayes classifiers also performs high-speed filtering. In the article filter, the bag-of-words method is used to represent text because we presume that some specific words or a simple combination of the words are enough to evaluate their PPI relevance at the article level, i.e. as a co-occurrence model.

Sentence filter

The sentence filter identifies PPI-related sentences from documents classified as relevant by the article filter. Since most of PPI sentences tend to have unique grammatical structures (7), a parse tree information which represents a set of words and its structural information is used to classify the PPI sentences. The convolution tree kernel in ref. (11) is adopted for calculating the similarity of grammatical tree structures without explicit rules or templates. The PPI-related sentences are obtained using the following procedure. First, input sentences are tagged by a rule-based part-of-speech (POS) tagger (12). The tagger is trained beforehand, using GENIA corpus (13). Second, the tagged sentences are parsed by a statistical natural language parser (14). Then, the parsing trees which do not have useful grammatical structures are discarded. After calculating sentence similarities by the tree kernel, the interaction patterns are predicted by SVMs (`http://www.csie.ntu.edu.tw/~cjlin/libsvm`). Finally, the probabilities of the PPI sentences are estimated using the SVM outputs.

USAGE

Figure 2 shows an example of using the PIE service; (A), (B) and (C) indicate input, PubMed search and output windows, respectively. In the full paper extraction process from uploaded files or copied text segments, (B) is skipped. (B) is only accessible when the 'PubMed Search' is clicked.

Input format

The PIE system accepts plain texts, HTML or PDF documents, as input. When an HTML or PDF document is given, built-in tools convert the document into plain text. The public tool (`http://www.foolabs.com/xpdf`), currently used for the PDF converting, may cause noisy text. For instance, if a PDF document has double-columned pages, the contents might be mixed up.

However, the support for PDF documents is necessary for general users because most papers are available as PDF format.

PubMed articles

PIE allows PubMed IDs as input to reduce efforts to enter texts or upload files. The 'PubMed Search' tool is provided for manually finding PubMed articles. When PubMed IDs are given, the corresponding abstracts are downloaded from online PubMed database. When keywords are given in the 'PubMed Search' tool, relevant abstracts are listed and users can select one or more abstracts to find PPI-related sentences. PIE uses tailored input by utilizing detailed article selection process, while other PPI services mostly conduct automatic PPI searching with a few keywords and predefined patterns. However, PIE also supports an automatic extraction method to obtain PPI information using a few keywords only. The 'I'm feeling lucky' button on the PubMed search tool performs this automatic extraction process.

All retrieved abstracts in search and output windows are linked to the actual paper pages in publishers' sites. Afterwards, one can use the downloaded papers for full paper extractions. Considering the lack of full paper database services, this feature might be useful.

Filter options

Three user options such as 'Tag Simplification,' 'Protein Dictionary' and 'Interaction Word Dictionary' are available for the sentence filter. 'Tag Simplification' transforms similar POS tags into representative one, i.e. NNS (noun, plural) and NNP (proper noun) are converted to NN (noun). Since most sentences in biomedical texts are syntactically complex, the tag simplification is necessary to reduce the structural complexity of the sentence. 'Protein Dictionary' and 'Interaction Word Dictionary' options use protein DB and interaction DB, respectively. The protein DB contains around 2.3 million protein words obtained from NCBI (`ftp://ftp.ncbi.nih.gov`). The interaction DB contains 1201 words, which is manually chosen by human experts based on the supplementary data in ref. (15). These options are used to incorporate the heuristic knowledge into the sentence filter.

Multiple session support

Users can maintain predicted PPI sentences by using session IDs. One can define a session in his or her own way and keep the PPI records using the session ID. Multiple sessions are allowed and identified by the session IDs. If PIE detects duplicate IDs, it shows 'APPEND' or 'NEW' buttons in the output window. 'APPEND' keeps

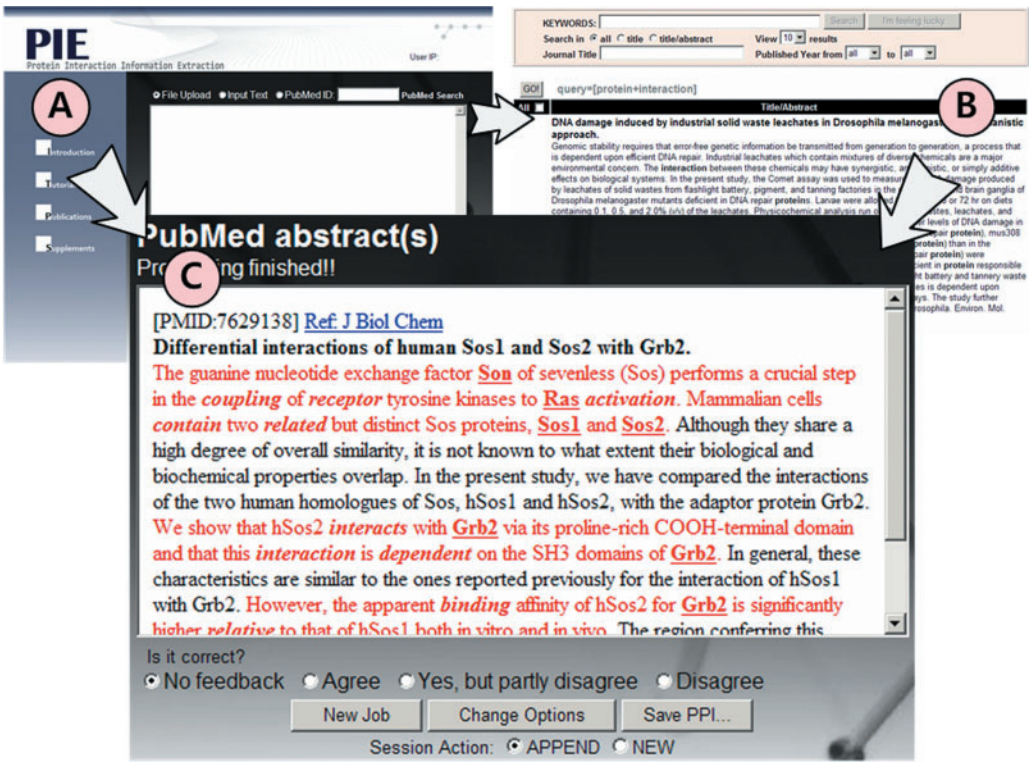

Figure 2. An example of PIE prediction results. PIE provides a user-friendly and intuitive interface. (**A**) Input. Web users can upload papers as a file or copy and paste text. A PubMed tool is provided for PubMed article searches. PIE allows multiple PubMed articles for PPI prediction in two ways, manual selection and automatic selection. (**B**) PubMed search. The article search using PubMed service is available for common use. The search results can be narrowed by the options such as number of results, published years and published journals. The 'I'm feeling lucky' button is for the automatic article selection, which does similar jobs as common PPI extraction tools do. (**C**) Output. Prediction results are listed in the center box, highlighting PPI sentences based on their probabilities. Colors of sentences represent their probabilities: 'Red' for high probability and 'Green' for moderate probability. According to the protein DB and the interaction DB, protein names and interaction-related words are indicated by bold and italic fonts, respectively. In particular, protein names are linked to the iHOP service for providing further information. Users can leave feedback to update PIE performance by selecting a 'No Feedback,' 'Agree,' 'Partly Disagree' or 'Disagree' button.

the current session, and 'NEW' restarts the session by deleting previous results. The multiple session concept is designed to save predicted PPI history in a local computer, and HTML is used to arrange the PPI history. Note that this is optional. If the Session ID remains as a blank, only current results are available for the file saving.

Output format

PPI sentences might appear in several places in a full document. Hence, PIE highlights the predicted PPI sentences on the original article, which improves user readability. Also, the system marks proteins and interaction-related words based on protein and interaction DBs, which helps biologists to identify the PPI information more directly. The protein words are retrieved only for the identified nouns by the natural language parser, and there are no such restrictions for the interaction-related words.

Detailed protein information is given by iHOP service. Highlighted proteins are linked to the search results of iHOP, which helps users to understand the protein functions in detail. Furthermore, predicted PPI sentences can be stored in a local computer for further use as already mentioned. The stored information can be utilized for literature summaries or curations for PPI database.

User feedback

To refine the performance of PIE, users can leave their feedbacks by marking a 'Agree,' 'Partly Disagree' or 'Disagree' button. These feedbacks are used to update our PPI extraction modules.

The training set of learning modules can differ according to domains. In such circumstances, PIE can be improved or customized depending on training data. The customized PIE is available upon user's request.

Remote procedure call support

The PIE system contains a running XML–RPC server. A client can send queries and receive the prediction results using the XML–RPC protocol. It provides more flexibility for using PIE. For instance, one can develop a meta service including PIE as a remote component. The XML-RPC specification is available at the PIE website.

RESULTS AND DISCUSSION

PIE is trained by the BioCreAtIvE II workshop dataset, enriched by Anne Lise Veuthey corpus, Prodisen interaction corpus and manually selected PPI sentence set (16). Using 10-fold cross-validation and 0.5 probability thresholds, the PPI article filter obtained 87.41% precision, 90.53% recall and 88.89% F1-score. The sentence filter obtained 92.13% precision, 91.78% recall and 91.96% F1-score.

The performance of PIE is evaluated on three different PPI corpus such as BioCreAtIvE I (BC) corpus (17), Christine Brun (CB) corpus (18), and N-PPI corpus (19). Since the PIE prediction outputs are the probabilities of examples, common precision and recall rates cannot be directly applied to evaluate the system performance. Therefore, PIE is evaluated using ROC (Receiver Operating Characteristic) curves and precision rates at *N*th ranked sentences. Among various options in PIE, the results with simplified tags and protein dictionary are shown in Figure 3, which depicts the ROC curves for the test data. In all cases, true-positive rate (TPR) is rapidly increased at low false-positive rate (FPR), which means that the system shows high precision rates for high probability sentences. More precisely, for top 30 ranked sentences, BC, CB and N-PPI show 83.87% precision, 96.77% precision, 70.97% precision, respectively.

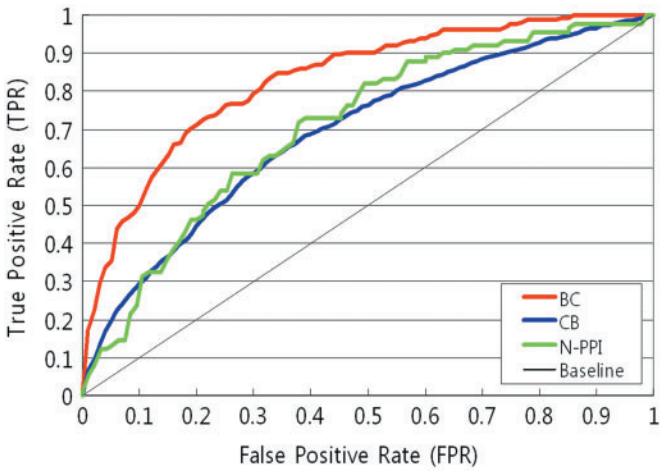

Figure 3. ROC curves for test data. Performance of PIE has been measured using independent test sets. The options on PIE was set to using simplified tags and protein dictionary. In all cases, TPR is rapidly increased at low FPR, implying that the system performs high precision predictions for high-probability sentences.

Our focus in the PIE system is to develop an ML-based framework for automatically identifying PPI sentences. This framework extends the availability of co-occurrence- and rule-based methods, and is able to find hidden patterns without predefined information. Subsequently, our approach reaches good precision rates for high probability sentences, which is one of the important properties for Web services. PIE is specialized to extract PPI sentences from text for summarizing or finding relevant information. Unlike other PPI services using keyword matching and predefined PPIs, the PIE interface handles user-provided full papers as well as PubMed articles online by utilizing ML properties. PIE does not use locally saved PPIs for system predictions, rather it utilizes online data obtained from users and other Web services. Thus, our system is more flexible to adopt new resources. If one wants to find PPI information initialized from few genes or proteins, other services such as iHOP would be a good choice. On the other hand, it is encouraged to use PIE for text-driven search derived from papers or keywords, particularly from newly published data.

In the current state, the PPI processing is a bit slow because of low parsing speed. The Collins parser used in PIE is well known, but old, which will be replaced with a faster tool near future. In addition, the preparsing for available PubMed articles would speed up the processing time in PIE, which remains as future work.

ACKNOWLEDGEMENTS

The authors would like to thank Jae-Hong Eom and Sung-Hwan Kim for inspiring their initial work. Mention of commercial products or services in this article does not imply approval or endorsement by NIST, nor does it imply that such products or services are necessarily the best available for the purpose. This work was supported by Korea Science and Engineering Foundation (M10400000349-06J0000-34910 to SK, IHL, SJK and BTZ); National Institute of Standards and Technology (the Manufacturing Metrology and standards for the Health Care Enterprise Program to SYS and RS); Korea Research Foundation (KRF-2006-214-D00140 to SYS). Funding to pay the Open Access publication charges for this article was provided by Seoul National University.

Conflict of interest statement. None declared.

REFERENCES

1. Cohen,A.M. and Hersh,W.R. (2005) A survey of current work in biomedical text mining. *Brief. Bioinform.*, **6**, 57–71.
2. Cases,I., Pisano,D., Andres,E., Carro,A., Fernández,J.M., Gómez-López,G., Rodriguez,J.M., Vera,J.F., Valencia,A. and Rojas,A.M. (2007) CARGO: a web portal to integrate customized biological information. *Nucleic Acids Res.*, **35**, W16–W20.
3. Krallinger,M. and Valencia,A. (2005) Text-mining and information-retrieval services for molecular biology. *Genome Biol.*, **6**, 224.
4. Chen,H. and Sharp,B.M. (2004) Content-rich biological network constructed by mining PubMed abstracts. *BMC Bioinformatics*, **5**, 147.
5. Jensen,L.J., Saric,J. and Bork,P. (2006) Literature mining for the biologist: from information retrieval to biological discovery. *Nat. Rev. Genet.*, **7**, 119–129.
6. Xiao,J., Su,J., Zhou,G.D. and Tan,C.L. (2005) Protein–protein interaction extraction: a supervised learning approach. In *Proceedings of the International Symposium on Semantic Mining in Biomedicine*. European Bioinformatics Institute, Hinxton, UK, pp. 51–59.
7. Jang,H., Lim,J., Lim,J.-H., Park,S.-J., Lee,K.-C. and Park,S.-H. (2006) Finding the evidence for protein–protein interactions from PubMed abstracts. *Bioinformatics*, **22**, e220–e226.
8. Hoffmann,R. and Valencia,A. (2004) A gene network for navigating the literature. *Nat. Genet.*, **36**, 664.
9. Fan,W., Stolfo,S., Zhang,J. and Chan,P. (1999) AdaCost: misclassification cost-sensitive boosting. In *Proceedings of the 16th International Conference on Machine Learning*. Morgan Kaufmann, San Francisco, USA, pp. 97–105.
10. Kim,Y.-H., Hahn,S.-Y. and Zhang,B.-T. (2000) Text filtering by boosting naive Bayes classifiers. In *Proceedings of the 23rd International ACM SIGIR Conference*. ACM Press, New York, USA, pp. 168–175.
11. Collins,M. and Duffy,N. (2001) Convolution kernels for natural languages. In *Proceedings of the 15th Conference on Neural Information Processing Systems*. Morgan Kaufmann, San Francisco, USA, pp. 625–632.
12. Brill,E. (1992) A simple rule-based part-of-speech tagger. In *Proceedings of the 3rd Conference on Applied Natural Language Processing*. Morgan Kaufmann, San Francisco, USA, pp. 151–155.
13. Kim,J.-D., Tomoko,O., Teteisi,Y. and Tsujii,J. (2003) GENIA corpus – a semantically annotated corpus for bio-textmining. *Bioinformatics*, **19(Suppl. 1)**, i180–i182.
14. Collins,M. (1999) Head-driven statistical models for natural language parsing. *PhD Thesis*. University of Penssylvania.
15. Hakenberg,J., Leser,U., Kirsch,H. and Rebholz-Schuhmann,D. (2006) Collecting a large corpus from all of Medline. In *Proceedings of the International Symposium on Semantic Mining in Biomedicine*. RWTH, Aachen, Germany, pp. 89–92.
16. Shin,S.-Y., Kim,S., Eom,J.-H., Zhang,B.-T. and Sriram,R. (2007) Identifying protein–protein interaction sentences using boosting and kernel methods. In *Proceedings of the 2nd BioCreAtIvE Workshop*. CNIO, Madrid, Spain, pp. 187–192.
17. Plake,C., Hakenberg,J. and Leser,U. (2005) Optimizing syntax patterns for discovering protein–protein interactions. In *Proceedings of the ACM Symposium on Applied Computing*. ACM Press, New York, USA, pp. 195–201.
18. Krallinger,M., Leitner,F. and Valencia,A. (2007) Assessment of the second BioCreative PPI task: automatic extraction of protein–protein interactions. In *Proceedings of the 2nd BioCreAtIvE Workshop*. CNIO, Madrid, Spain, pp. 41–54.
19. Sanchez-Graillet,O. and Poesio,M. (2007) Negation of protein–protein interactions: analysis and extraction. *Bioinformatics,* **23**, i424–i432.