European Journal
of Medical Research

RESEARCH

Open Access

Biomedical literature mining: graph kernel-based learning for gene–gene interaction extraction

Ai-Ru Hsieh[1*] and Chen-Yu Tsai[1]

Abstract

The supervised machine learning method is often used for biomedical relationship extraction. The disadvantage is that it requires much time and money to manually establish an annotated dataset. Based on distant supervision, the knowledge base is combined with the corpus, thus, the training corpus can be automatically annotated. As many biomedical databases provide knowledge bases for study with a limited number of annotated corpora, this method is practical in biomedicine. The clinical significance of each patient's genetic makeup can be understood based on the healthcare provider's genetic database. Unfortunately, the lack of previous biomedical relationship extraction studies focuses on gene–gene interaction. The main purpose of this study is to develop extraction methods for gene–gene interactions that can help explain the heritability of human complex diseases. This study referred to the information on gene–gene interactions in the KEGG PATHWAY database, the abstracts in PubMed were adopted to generate the training sample set, and the graph kernel method was adopted to extract gene–gene interactions. The best assessment result was an F1-score of 0.79. Our developed distant supervision method automatically finds sentences through the corpus without manual labeling for extracting gene–gene interactions, which can effectively reduce the time cost for manual annotation data; moreover, the relationship extraction method based on a graph kernel can be successfully applied to extract gene–gene interactions. In this way, the results of this study are expected to help achieve precision medicine.

Keywords Gene–gene interaction, Graph kernel, Biomedical text mining

Introduction

Text mining of scholarly literature for the purpose of information extraction is an effective method to keep up with the latest research findings. The first step of information extraction is called entity recognition (NER) [20]. Biomedical NER aims to identify related entities in text, such as proteins, genes, or diseases.

With the development of the internet, the amount of biomedical literature citations is increasing exponentially; for example, more than 30 million biomedical literature citations are in PubMed, which contains a large amount of biomedical knowledge. However, biomedical researchers have difficulties in obtaining the desired information from this huge data in a timely manner. One of the solutions to obtain the desired information from the huge biomedical literature citations is biomedical relationship extraction. For biomedical research, gene–gene interactions can help explain the heritability of human complex diseases.

Accounting for the effect of genetic interactions can help with detecting gene functions and pathways. These

*Correspondence:
Ai-Ru Hsieh
airudropbox@gmail.com
[1] Department of Statistics, Tamkang University, Tamsui District, New Taipei City 251301, Taiwan

interactions include suppressive, synthetic, and epistatic types. One of the most famous strategies for detecting gene–gene interaction is the multifactor dimensionality reduction (MDR) method [6, 34] which has been used profusely. The machine learning (ML) method is a powerful alternative to traditional methods for analyzing gene–gene interactions [7, 8, 11, 12, 24, 38, 40, 51]. In the development of new gene–gene interaction extraction methods, previous literature is required for validation. The biomedical relationship extraction system can achieve this target. GeneDive [33, 49] and DeepDive [28] are biomedical relationship extraction systems. GeneDive is a text-mining method that can search, sort, group, filter, highlight, and visualize interactions between drugs, genes, and diseases. DeepDive is a probabilistic inference system that uses factor graphs to calculate the probabilities of random variables. There is also a gene–gene interaction literature mining on Host-Brucella [22]. Protein–protein interactions (PPIs) represent a fundamental form of molecular interaction that governs critical biological functions such as cell proliferation, differentiation, and apoptosis. STRING [13] is a text mining-based PPI method that conducts statistical co-citation analysis across many scientific texts, including all PubMed abstracts. Many human PPI research studies were retrieved from STRING (https://string-db.org/, version 9.1) [41, 48, 50, 52].

Supervised machine learning has been widely employed in relationship extraction [1]. As relationship extraction is regarded as a classification problem, effective features are designed according to the annotated corpus to learn different classification models, and then, a trained classifier is adopted to predict relationships (such as the positive or negative relationship of gene–gene interactions in KEGG PATHWAY). In this method, an annotated corpus is required to learn and predict labels of new instances,therefore, if the supervised machine learning method is employed to train the relationship extraction system, a corpus is necessary. However, the corpus requires a lot of manual tagging, which is a problem, as it is time-consuming and labor-intensive. Unfortunately, there is a limited number of corpora for gene–gene interactions in genetics. PubTator allows curators to specify the kind of relations they desire to capture from the literature, which can be either between the same kind of entities, such as protein–protein interactions, or between different kinds, such as gene–disease relations [45]. PubTator entity recognition tools include GeneTUKit [18] for gene mention, GenNorm [43] for gene normalization, SR4GN [44] for species, tmVar [42] for mutations and a dictionary-based lookup approach [47] for chemicals.

A graph kernel is a kind of kernel function in which graphs are input, and then the similarities of graph pairs are output. There are two methods to apply graph kernel to relationship extraction. One method is that a kernel function is constructed according to the text features [dependent graph, and part of speech (POS)], and the kernel function (APGK and ASMK) is adopted to calculate the distance between two relationships [1, 31]. The other method is that graph features are constructed according to text features (i.e., dependent graph, and POS), and the kernel function (i.e., Random Walk Kernel, and Labeled Graph Kernel) is employed to calculate the distance between two relationships according to the structure of the graph features [53].

There are two main research objectives in this study. First, due to the multiple gene–gene interactions, it contains [21], the KEGG PATHWAY database was taken as the knowledge base, and distant supervision was carried out to automatically generate a large number of labeled data, to solve the problem of insufficient training data for gene–gene interactions in genetics. Second, a machine learning method based on the graph kernel was proposed to extract gene–gene interaction relationships in the text. The distant supervision method, as developed in this study, can automatically create a corpus for extracting gene–gene interactions, which can effectively reduce the time cost for manual annotation data, moreover, the relationship extraction method based on a graph kernel can be successfully applied to extract gene–gene interactions.

Distant supervision is an extension of the paradigm used by WordNet to extract hypernym relationships between entities, similar to using weakly labeled data in bioinformatics [9, 29]. Some of the commonly extracted biomedical relations are microRNA–gene relations [25], protein–protein interactions [32], and disease–gene relationships [23]. However, the practice of precision medicine will ultimately require healthcare providers to refer to gene and mutation databases to understand the clinical significance of each patient's genetic makeup [37]. Unfortunately, there is a lack of previous biomedical relationship extraction research in gene–gene interaction. Our study developed a distant supervision method to automatically create a corpus for extracting gene–gene interactions. Then, we used a graph kernel ML approach to extract gene–gene interaction relationships in the text. In this way, our proposed method is expected to help achieve precision medicine.

Materials and methods

Corpus

Our study obtained abstracts from PubMed, based on references of KEGG and MeSH keywords. For KEGG, we selected 93 KEGG PATHWAY (http://www.kegg.jp/ or http://www.genome.jp/kegg/) belonging to human diseases and then mapped them to a PubMed ID. For MeSH, we employed the NCBI database (https://www.ncbi.nlm.

nih.gov/) using Medical Subject Heading (MeSH) terms and query the keyword of [genetic interaction]. Based on the query, our study obtained "interaction" medical terms (i.e., "epistatic, genetic", "genetic susceptibility disease"), and "disease" and "genetic" medical terms (i.e., "disease" "Genetic"). Then, we filtered abstracts from PubMed for the years (1946–2021) based on the above keywords.

The final corpus consists of 132,946 abstracts. Among them, 2,071 abstracts are from KEGG PATHWAY and 130,871 abstracts are from MeSH.

Text preprocessing

There are main three steps in text preprocessing: In Step 1, PubTator Central (PTC) [45] was adopted for Name Entity Recognition (NER) to recognize genes in text abstracts [45]. In Step 2, the natural language model of ScispaCy (i.e., en_core_sci_scibert) was employed to break the abstract and take the sentences of gene–gene co-occurrence. In Step 3, the KEGG PATHWAY was taken as the knowledge database to identify the relationship category of sentences, and a training corpus was constructed for distant supervision based on this relationship category. Each data in the corpus contains a sentence with two genes, the article source of the sentence, the NCBI ID of the two genes, and whether there is an interaction in KEGG.

Graph feature construction

The ScispaCy model was adopted to conduct dependency analysis and part-of-speech tagging, and then, these features were employed to construct training biomedical relations for the input classifier.

Tokenization and dependency parsing

For all the features extracted from the corpus under distant supervision, including syntactic dependency parsing and part-of-speech tagging, the natural language model of ScispaCy (i.e., en_core_sci_scibert) was adopted, which contains approximately 785,000 biomedical vocabularies. Step 1 was tokenization, and the Tokenizer of spaCy was employed for tokenization. Step 2 was part-of-speech tagging, and the Tagger of spaCy was adopted to extract the part-of-speech labels of all tokens except gene words. Step 3 was dependency parsing, where the parser of spaCy was employed to identify the relationship between words in sentences and obtain the dependency relation between words.

Sentence category determination

To train the classifier, after extracting the sentence containing two genes, it is also necessary to obtain the category of the sentence. The sentence category determination in this study can be divided into three steps: In Step 1, direct and indirect gene–gene interactions were obtained from the KEGG PATHWAY database. In Step 2, all possible gene pairs in the sentence were obtained. Since a sentence may contain multiple gene entities, all sentences were arranged and combined to list all possible gene pairs. In Step 3, whether the gene pairs belong to the gene–gene interactions from KEGG PATHWAY was determined, if yes, a positive category was given, otherwise a negative category was given. The steps in 2.3.2 were performed in Python 3.6.5.

Graph-based features

After the features were obtained through tokenization and dependency parsing, NeworkX [15] was adopted to construct the graphs, and the sentences were structured as the input of the classifier. During graph construction, according to the method introduced by Bunescu and Mooney [4], a weighted graph was constructed for each sample of the corpus. The vertex in the weighted graph was composed of tokens in a sentence, with POS labels added,the edge was composed of the dependency relation between tokens, with attributes added. During graph feature construction, basic syntactic features (including dependency graph, and POS) were employed to enrich the graphs in combination with heuristic methods [including direction and shortest dependency path (SDP)]. Therefore, the graph kernel can contain more information during the calculation of the similarities between graphs.

According to the shortest path hypothesis, as proposed by Bunescu and Mooney [4], the shortest dependency path (SDP) of two entities contains the most information in a sentence. Therefore, when the edge weight is added, concerning the weight configuration put forward by Airola et al. [1], the edge belonging to the SDP was given a weight of 0.9, while the remaining were given a weight of 0.3. The semantic relationships between nouns were classified according to SemEval 2010 relationship identification [16].

Our study adopted the directional relationship and the dependency relation as the attributes to construct the edges of the NetworkX graphs. Finally, for each sentence sample, a total of four features were selected in this study to construct the NetworkX graphic data, including POS, dependency relation, directional relationship, and SDP. An example of initialization token vector representations is shown in the Supplementary Fig. s4. The dependency relation is all binary: a grammatical relation holds between a governor (a regent or a head) and a dependent. The grammatical relations are defined in Marneffe [27]. Detailed descriptions of the various dependency types are available in the Stanford Typed Dependencies manual provided by the Stanford NLP [27]. The directional relationships could be divided by forward and backward parsing. In forward parsing, the input tokens are in their original order, while in backward parsing, the input tokens are from right to left [5]. We used

"These agents negatively regulate the Gene_3667 (IRS1) functions by phosphorylation but also via others molecular mechanisms (Gene_1154 (SOCS) expression, Gene_3376 (IRS) degradation, O-linked glycosylation) as summarized in this review." as an example sentence (Fig. s2).

SVM graph kernel

The constructed feature graphs were classified by the SVM graph kernel method to extract the relationships. According to Borgwardt et al. [2], from GreKel [36], three commonly adopted kernel methods were selected, including Shortest Path [30], Weisfeiler–Lehman subtree [35], and Neighborhood Hash [17]. For the following three graph kernel functions, the graph feature (G) of a sample was defined to consist of a set of the vertex (V) and the edge (E): $G = (V, E)$, where each edge is in the form of $e = (u, v)$, and $u, v \in V$.

Shortest path kernel (SPK)

The shortest path kernel (SPK), as proposed by [3], can compare the similarities of two graphs according to the SDP features (length and vertex label).

Let G_1, G_2 be two feature graphs. First, according to Dijkstra's Algorithm [10], the SDPs between all vertex pairs were calculated, and G_i, G_j were transformed into shortest path graphs S_1, S_2, where $S_1 = (V_1, E\prime_1), S_2 = (V_2, E\prime_2), E\prime \subseteq E$. Then, the similarity of two shortest dependency paths (SDPs) was calculated to obtain the definition of the SPK, as Eq. (1.1):

$$k_{SP}(S_1, S_2) = \sum\nolimits_{e_1 \in E_1} \sum\nolimits_{e_2 \in E_2} k_{walk}^{(1)}(e_1, e_2) \quad (1.1)$$

where $k_{walk}^{(1)}(e_1, e_2)$ is the positive definite kernel on the edge walk with length 1 in the shortest path graph, which is employed to assess the similarity of paths. Given two edges $e_1 = \{u_1, v_1\}$ and $e_2 = \{u_2, v_2\}$, $k_{walk}^{(1)}(e_1, e_2)$ is defined as the inner product of three Dirac kernels, including two vertex kernels (k_{vertex}) for comparing the beginning and end of the path and an edge kernel (k_{edge}) for comparing the length of the SDP, as Eq. (1.2):

$$k_{walk}^{(1)}(e_1, e_2) = k_{vertex}(u_1, u_2) \times k_{edge}(e_1, e_2) \times k_{vertex}(v_1, v_2) \quad (1.2)$$

Weisfeiler–Lehman Kernel (WLK)

The Weisfeiler–Lehman kernel framework [35] is a kernel framework based on the Weisfeiler–Lehman algorithm [46]. This study adopted the vertex histogram kernel [39].

The Weisfeiler–Lehman algorithm is a method for label refinement, and the main steps are as follows:

1. Give an initial input graph $G = (V, E)$, and vertexes with labels.
2. A multi-label set was constructed for the label of each vertex in the graph, as composed of the vertex and its neighbors.
3. The obtained multi-label set was compressed into a new label to re-label the vertex as a new label.
4. The processes of 2 and 3 were repeated for h times, where h was the set iteration number.

When there is more than one input graph, these steps would be simultaneously performed for all the input graphs. If the vertexes from different graphs have the same multi-label set, they would have the same new label.

The Weisfeiler–Lehman kernel requires a graph kernel as the iteration benchmark. Let k be the vertex histogram kernel, which is called the base kernel. The vertex histogram kernel would judge the similarity of graphs according to the number of vertexes and labels of vertexes. Assuming that every vertex of a graph set comes from an abstract vertex space V, it is given a set of vertex labels $\boldsymbol{L}, l : \boldsymbol{V} - > \boldsymbol{L}$ as the label assignment function for the vertexes. If the total number of labels is d, that is, $d = |\boldsymbol{L}|$, then graph G can be expressed as vector $\mathbf{s}$, $\mathbf{s} = (s_1, s_2, ..., s_d), s_i = \left|\left\{v \in V : l(v) = i\right\}\right|, i \in L.$

Let G_1, G_2 be two feature graphs, then the vertex histogram kernel is defined as Eq. (2.1):

$$k(G_1, G_2) = \langle \mathbf{s}_1 , \mathbf{s}_2 \rangle \quad (2.1)$$

Let G_1, G_2 be two feature graphs, and the Weisfeiler–Lehman Kernel taking k as the benchmark kernel is defined as Eq. (2.2):

$$k_{WL}(G_1, G_2) = k(G_{10}, G_{20}) + k(G_{11}, G_{21}) + ... + k(G_{1h}, G_{2h}) \quad (2.2)$$

where h is the iteration number of Weisfeiler–Lehman. In this study, h was set as 5, and $\{G_{10}, G_{11}, ..., G_{15}\}$ and $\{G_{20}, G_{21}, ..., G_{25}\}$ were the Weisfeiler–Lehman sequences generated by G_1 and G_2 by iteration, respectively.

Neighborhood hash kernel (NHK)

In terms of the neighborhood hash kernel (NHK), the similarity of graphs was calculated by updating the labels of the vertexes and calculating the number of common labels. Zhang et al. applied the neighborhood hash kernel (NHK) in biomedical literature exploration to extract protein–protein interactions, which had a good effect on the dependency graph [53]. This study applied the neighborhood hash kernel (NHK) to extract gene–gene interactions. Let G_1, G_2 be two feature graphs, the hash labels of the two graphs were calculated according to the neighborhood hash function, and the NHK was defined to compare the similarities, as Eq. (2.3):

$$k_{NH}(G_1, G_2) = \frac{b}{a_1 + a_2 - b} \quad (2.3)$$

where b represents the number of labels shared by two graphs, and a_1,a_2 represent the number of vertexes of G_1, G_2, respectively.

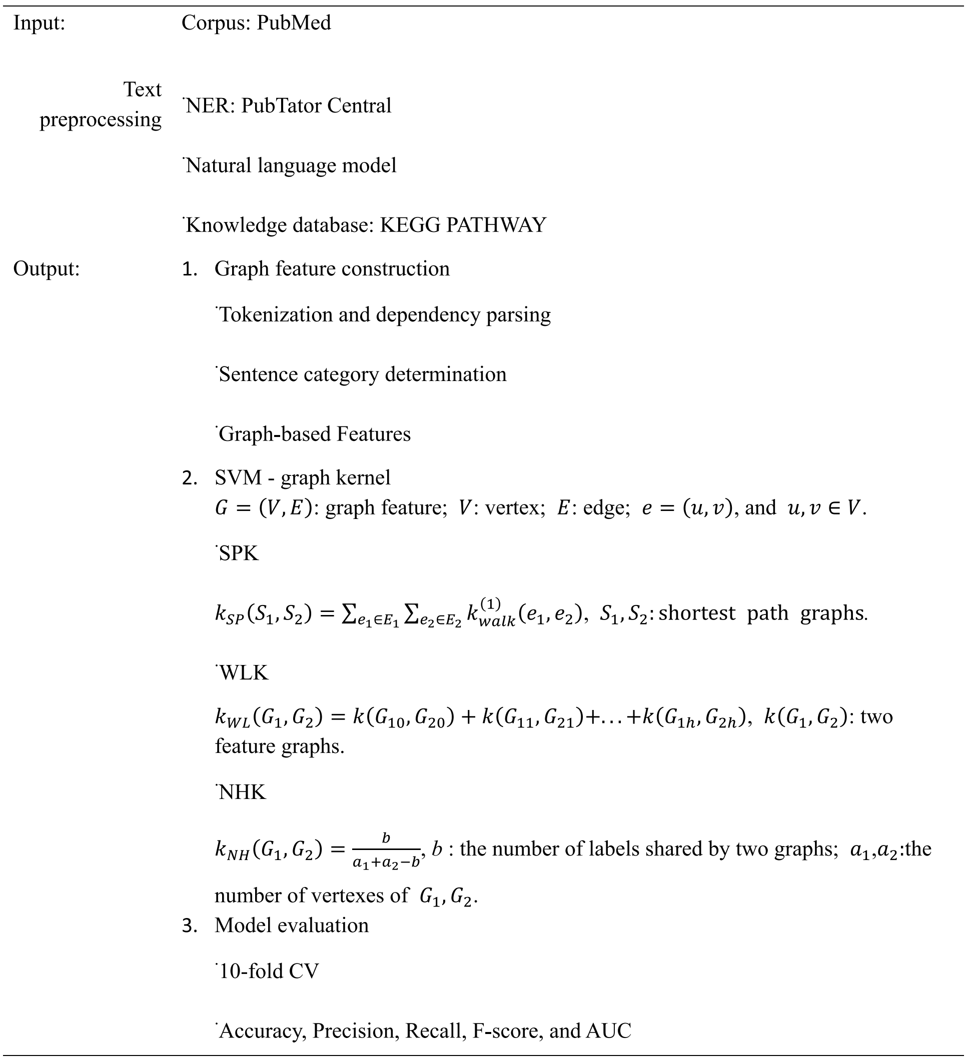
CV and performance evaluation

This study employed KEGG PATHWAY as the knowledge base for distant supervision, and the final corpus contained 392,369 data, including 10,500 positive cases (i.e., gene–gene interactions in KEGG PATHWAY) and 381,869 negative cases (i.e., non-gene–gene interactions in KEGG PATHWAY). To avoid over-training data, which would result in overfitting, tenfold cross-validation was adopted to train the classifier. However, due to an excessive difference between the positive and negative ratio (1:40) of the data, the data could not be directly cut into tenfold for tenfold cross-validation. Therefore, random sampling was conducted for the positive and negative samples, respectively, and then, the removed samples were not put back, to ensure the independence of the samples. The positive and negative sampling ratio was 1:3, and the final dataset was divided into tenfold, each with 4,200 samples (Fig. s3).

Then, tenfold cross-validation was carried out according to the above dataset division (Fig s3). With 9/10 as the training set and the rest 1/10 set as the testing set, three different graph kernels (SPK, WLK, and NHK) were employed to train the SVM classifier. The iteration number of the WLK updating labels was set to five times, and the maximum neighbor number of NHK single labels was three. All the above experimental processes were performed on Python.

To assess the results for the relationship extraction classification task completed by the three graph kernels, a confusion matrix was established, and several common indicators were adopted for the assessment, including Accuracy, Precision, Recall, F1-score, and area under curve (AUC).

Algorithm



```
Input:              Corpus: PubMed

Text preprocessing  ˙NER: PubTator Central
                    ˙Natural language model
                    ˙Knowledge database: KEGG PATHWAY

Output:             1. Graph feature construction
                       ˙Tokenization and dependency parsing
                       ˙Sentence category determination
                       ˙Graph-based Features
                    2. SVM - graph kernel
                       $G = (V, E)$: graph feature; $V$: vertex; $E$: edge; $e = (u, v)$, and $u, v \in V$.
                       ˙SPK
                       $k_{SP}(S_1, S_2) = \sum_{e_1 \in E_1} \sum_{e_2 \in E_2} k_{walk}^{(1)}(e_1, e_2)$, $S_1, S_2$: shortest path graphs.
                       ˙WLK
                       $k_{WL}(G_1, G_2) = k(G_{10}, G_{20}) + k(G_{11}, G_{21}) + \ldots + k(G_{1h}, G_{2h})$, $k(G_1, G_2)$: two feature graphs.
                       ˙NHK
                       $k_{NH}(G_1, G_2) = \frac{b}{a_1 + a_2 - b}$, $b$ : the number of labels shared by two graphs; $a_1$,$a_2$:the number of vertexes of $G_1, G_2$.
                    3. Model evaluation
                       ˙10-fold CV
                       ˙Accuracy, Precision, Recall, F-score, and AUC
```

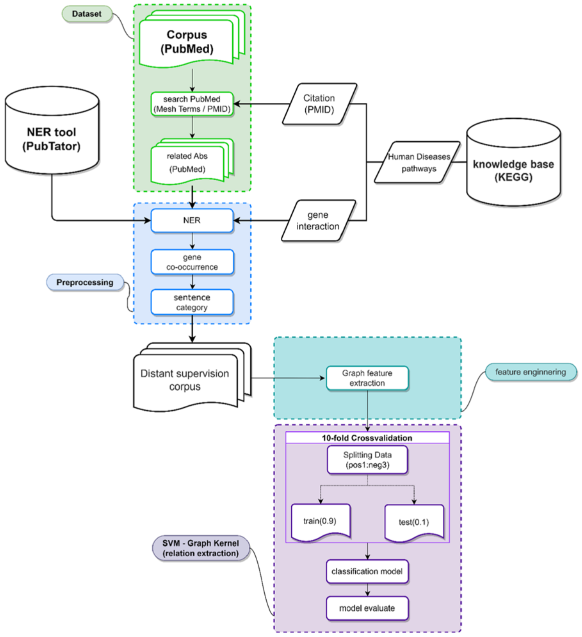


Fig. 1 Flow diagram showing the process of analysis for this study

Results

A flow chart of the analysis process for this study is shown in Fig. 1. An assessment was conducted on the results of the method constructed in this study for extracting gene–gene interactions. Then, the data were described, and the assessment results of the distant supervision relationship corpus employed for the three graph kernels were discussed.

10-fold CV

Figure 2 shows the results of tenfold CV using three graph kernels on SVM. Regarding the results of tenfold CV in the training set, the accuracy of SPK was 0.7518–0.7528, the accuracy of WLK was 0.8122–0.8148, and the accuracy of NHK was 0.8331–0.8385. Compared with those in the training set, regarding the results of tenfold CV in the testing set, the accuracy of SPK decreased by approximately 0.001 as 0.7505–0.7526; the accuracy of WLK decreased by approximately 0.03 as 0.7674–0.7819; the accuracy of NHK decreased by approximately 0.05 as 0.7764–0.7917. According to the results, regarding the difference between the accuracy of the model in the training set and that in the testing set, SPK was the smallest, followed by WLK and NHK, respectively.

Classification results

An assessment was made on the interaction corpus, as obtained by distant supervision through the SVM graph kernel classification, and the results show that

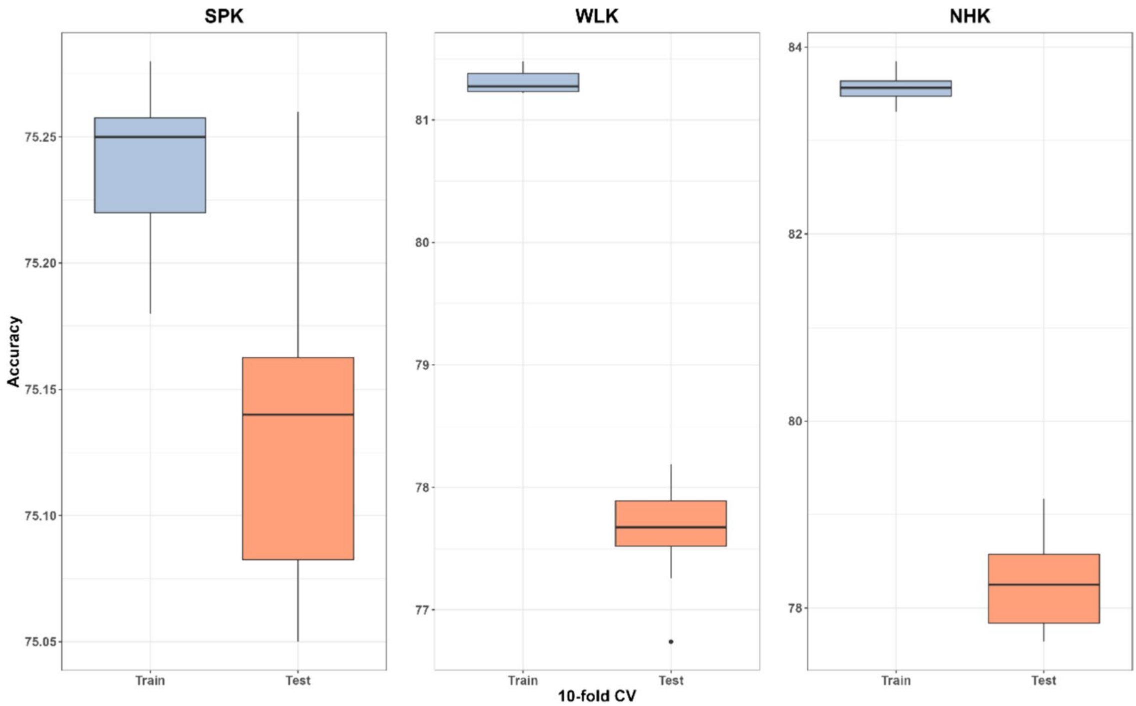


Fig. 2 The results of tenfold CV using three graph kernels methods in shortest path kernel (SPK), Weisfeiler–Lehman kernel, (WLK) and neighborhood hash kernel (NHK). The accuracy of training data with (blue) and testing data (orange) is shown

Table 1 Classification results from SPK, WLK, and NHK

Graph kernel	Accuracy	Precision	Recall	F-score
Shortest path	0.75	0.81	0.75	0.75
Weisfeiler–Lehman	0.78	0.78	0.78	0.78
Neighborhood hash	0.79	0.79	0.79	0.79

all the three graph kernels had good classification efficiency (F-score > 70%, Table 1). According to Table 1, regarding the performance of accuracy, NHK was the best (Acc = 0.79), followed by WLK (Acc = 0.78) and SPK (Acc = 0.75), respectively. The same trends were shown in the performances of Recall and F-score (NHK (Recall = 0.79, F-score = 0.79), WLK (Recall = 0.78, F-score = 0.78), and SPK (Recall = 0.75, F-score = 0.75). According to Table 1, regarding the performance of F-score, SPK was poor (F-score = 0.75) because vertex labels and vertex attributes were adopted for SPK to calculate the similarity without considering edge labels; compared with SPK, NHK (F-score = 0.79) and WLK (F-score = 0.78) had better performance because more information on the edge labels was employed. In addition, due to a small number of TP in SPK, SPK had the highest value in precision ($P = 0.81$); however, many samples with actual interactions were ignored for that reason. Finally, the McNemar test P value of WLK and NHK was $3 \times e^{-8}$, and the two classifiers, WLK and NHK, had a significant statistical difference in classification accuracy. In terms of accuracy, NHK had a better performance than WLK. Figure 3 shows the AUC values of the three graph kernel methods, which were all between 0.7 and 0.8, and WLK (AUC = 0.77) and NHK (AUC = 0.76) had better classification results than SPK (AUC = 0.71).

NHK and WLK had the best classification results in F1-score and AUC. In terms of calculation time, although a simple benchmark kernel was adopted for WLK, the calculation time of WLK was much lower than that of NHK. However, since the positive and negative ratio of the data was 1:3, the data was unbalanced. Therefore, the F-score had a better assessment effect than AUC. We suggest that, in the case of unbalanced data and plenty of time, NHK can achieve better classification results; in the case of a limited time, WLK can obtain classification results not inferior to NHK.

The data obtained by distant supervision were mostly unbalanced, thus, the positive and negative ratios of the dataset were further divided into two other settings (1:10, 1:100) for comparison. According to the results, the greater is the difference between positive and negative ratios, the worse the prediction effect (P, R, F) of positive cases (Fig. 4). We remind hereby that if the data obtained by the distant supervision method is employed for machine learning, the sampling is required to solve the problem of unbalanced samples.

Discussion

This study put forward a distant supervision method, in which the KEGG PATHWAY database was employed as the knowledge base, and 132,946 abstracts from PubMed were tagged, to obtain annotated training data. According to the results, the annotated datasets created by the three graph kernel methods could be effectively adopted as a training sample for machine learning to extract gene–gene interactions from the KEGG PATHWAY database.

The dataset employed in this study was the abstracts taken from PubMed. Since an abstract usually only

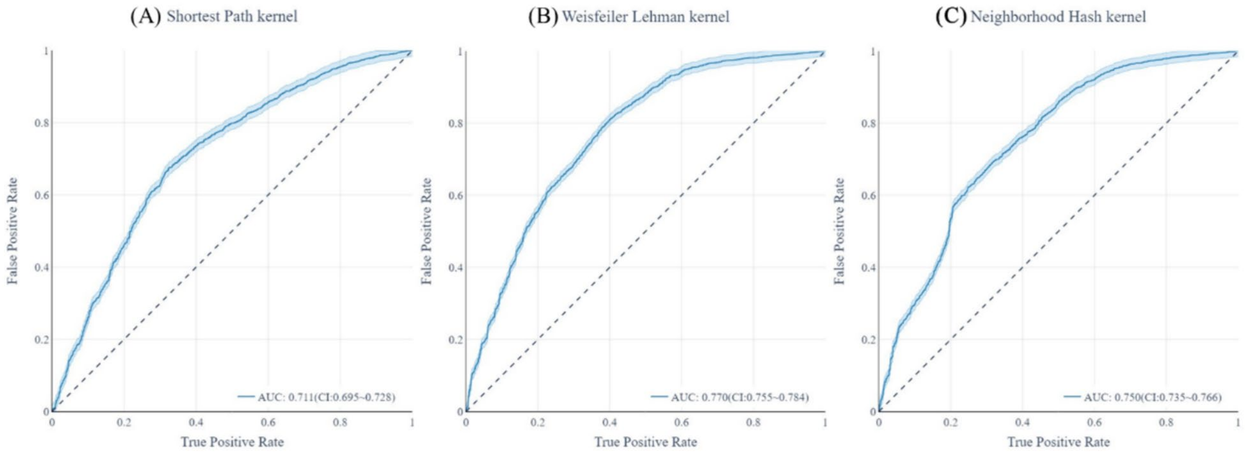


Fig. 3 Receiver-operating characteristic curves for the three graph kernel methods in **A** shortest path kernel (SPK), **B** Weisfeiler–Lehman kernel, WLK, and **C** neighborhood hash kernel (NHK)

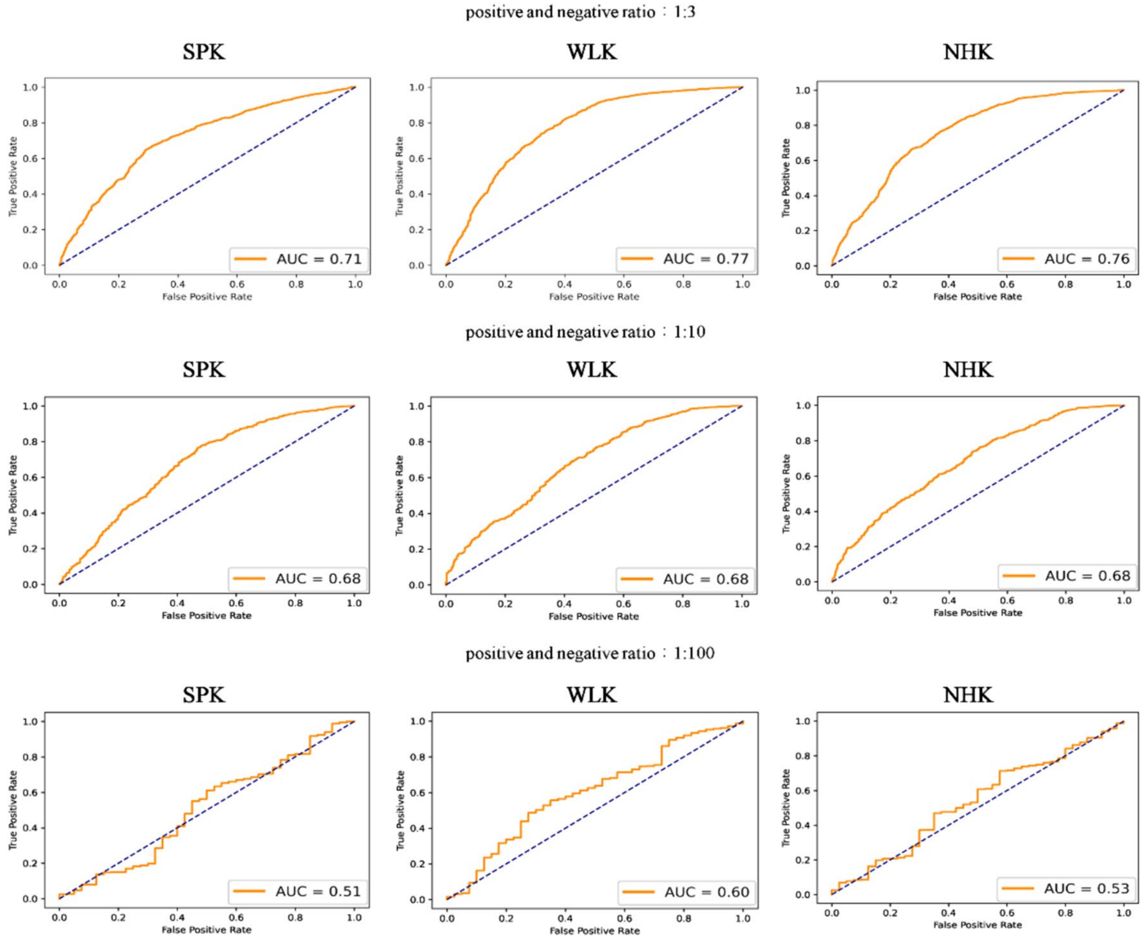


Fig. 4 Receiver-operating characteristic curves for the three graph kernel methods in shortest path kernel (SPK), Weisfeiler–Lehman kernel (WLK), and neighborhood hash kernel (NHK) on the three positive and negative ratios (1:3, 1:10, and 1:100)

describes newly discovered gene–gene interactions, the gene–gene interactions published in the past may not appear in the abstract of a text, thus, the classification results of the three graph kernel methods may be underestimated. Therefore, in this study, the abstracts over a long period of years (1946–2021) were captured from PubMed to solve this problem.

Three graph kernels were applied to the constructed feature graph. As adopted by these graph kernels, the label information of graphs included POS, dependency relation, and directional relationship. In terms of the shortest dependency path (SDP) feature that was not employed, analysis was also conducted on the random walk kernel where the attribute information could be adopted. However, due to the consideration of attribute information, the calculation of this graph kernel required a lot of computer resources, thus, due to the limitation of hardware equipment, only 400 data were included in the analysis in this study. According to the results, the F1-score of the graph feature in the random walk kernel reached 0.86 (Table s2), the accuracy was 0.75 (Table s2), and the AUC was 0.57 (Fig. s1).

GeneDive [33, 49] is a single-page web application following the widely used model-view-controller (MVC) architecture. STRING [13] is a text mining-based PPI method. We analyzed *VHL* and *CREBBP* genes by GeneDive and STRING. The supplementary shows the *VHL* and *CREBBP* interaction results (Fig. s5, Fig. s6, Table s3, and Table s4). In addition to GeneDive and STRING, our method can provide an alternative method to gene–gene interaction text extraction. Using GeneDive and STRING, we can ensure the reliability and accuracy of our result findings.

In practice, the resulting dimensionality of the space is usually very large, since the number of dimensions is determined by the number of distinct index terms in the corpus. Therefore, techniques to control the dimensionality of the vector space are often required. Due to the superior potential of SVM techniques in efficiently managing high-dimensional input spaces compared to other classification techniques, the need for time-consuming language preprocessing (i.e., reducing the dimensionality of feature spaces) can be largely eliminated. Therefore, we choose SVM methods for classification in this study.

The KEGG PATHWAYS or modules are represented as lists of genes, which can be obtained from the literature or online repositories such as Gene Ontology and KEGG. The KEGG PATHWAY modules use z-tests to determine the relative importance of the corresponding modules or pathways in different patient groups [14, 26]. A module is "positively enriched" in a sample if it has a positive z-score with a corrected $P < 0.05$ and is "negatively enriched" if the z-score is negative with a corrected $P < 0.05$[14, 19, 26].

The method constructed in this study is limited to using KEGG PATHWAY as the knowledge base. Although the KEGG PATHWAY database contains most gene–gene interactions, there are still gene–gene interactions that are not included in KEGG PATHWAY, which may not be identified by this relationship corpus. In the future, more databases with gene–gene interactions other than KEGG PATHWAY are expected to be adopted as the knowledge base for distant supervision to create a richer corpus.

The distinction between KEGG and PubMed as data sources is significant, and we consider that varying confidence scores should represent the disparity in their quality. This study primarily utilizes references from KEGG and MeSH keywords to analyze abstract data procured from PubMed. The keywords within these abstracts are derived from KEGG and MeSH. We will further incorporate the assessment of confidence scores in our subsequent research.

Zhang et al. [53] proposed a neighborhood hash graph kernel method, and Zhang et al. [54] proposed a hash subgraph pairwise kernel-based method for extraction of protein–protein interactions from biomedical literature. These two previously proposed methods are similar to ours. Unfortunately, their methods are only for the protein–protein interaction corpus and cannot be used for our gene–gene interaction corpus. Therefore, it is hard to make a comparison with our findings. The limitation of our study is that the corpus of gene–gene interaction is very scarce. In this study, we used tenfold CV to increase the stability of our proposed methodology. In future studies, we will try to find different gene–gene interaction corpora to improve the performance of our method.

The SVM graph kernel method and random sampling were employed to assess the distant supervision corpus. Although the feasibility of the distant supervision method has been proven, as there is an enormous amount of data in the distant supervision corpus, when there is enough data and time, the effect of deep learning will usually be better than that of SVM. In the future, we anticipate that this distant supervision corpus can be trained through deep learning to identify gene–gene interactions from the literature directly.

Conclusion

In this research, we have developed and implemented a novel distant supervision approach for the automatic creation of a corpus specifically designed for the extraction of gene–gene interactions. Our methodology leverages a graph kernel machine learning (ML) technique to effectively identify and extract these complex relationships within textual data. The implications of our work are significant, offering a robust tool for researchers and professionals in the field of genomics. By automating the extraction process and enhancing the accuracy of gene–gene interaction identification, our proposed method stands to substantially contribute to the advancement of helping to understand human diseases.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s40001-024-01983-5.

Supplementary material 1: Fig. S1. Receiver-Operating Characteristic curve for random walk kernel. Fig. S2. Graph kernel example. Fig. S3. Flow diagram showing the process of 10-fold CV for this study. Fig. S4. An example of initialization token vector representations. Fig. S5. The *VHL* and *CREBBP* interaction results by GeneDive. Fig. S6. The *VHL* and *CREBBP* interaction results by STRING. Table S1. The statistics of the generated corpus. Table S2. The F1-score and accuracy in the Random Walk Kernel. Table S3. The enrichment analysis mainly focuses on *CREBBP* by STRING. Table S4. The enrichment analysis mainly focuses on *VHL* by STRING.

Acknowledgements

We acknowledge the staffs of Taiwan biobank for their hard work in collecting and distributing the data. This work was supported by the following grants: NST 110-2314-B-032-001 and 112-2314-B-032-001. We thank the Department of Statistics, Tamkang University of Taiwan, and National Science and Technology Council in Taiwan.

Author contributions

For research articles with several authors, a short paragraph specifying their individual contributions must be provided. The following statements should be used "Conceptualization, A.-R. H. and C.-Y. T.; methodology, A.-R. H. and C.-Y. T.; software, C.-Y. T.; validation, A.-R. H.; formal analysis, C.-Y. T.; investigation, A.-R. H.; resources, A.-R. H.; data curation, A.-R. H. and C.-Y. T.; writing—original draft preparation, A.-R. H. and C.-Y. T; writing—review and editing, A.-R. H.; visualization, A.-R. H.; supervision, A.-R. H.; project administration, A.-R. H.; funding acquisition, A.-R. H.

Funding

This research was funded by the National Science and Technology Council in Taiwan (NST 110-2314-B-032 -001 and 112-2314-B-032-001).

Availability of data and materials

All analysis results can be shared on request by contacting the corresponding authors for reasonable use.

Declarations

Ethics approval and consent to participate

The study was conducted according to the guidelines of the Declaration of Helsinki and approved by the Institutional Review Board of Taipei Veterans General Hospital (ID No: 2020-12-009AC).

Informed consent

Informed consent was obtained from all the subjects involved in the study.

Competing interests

All the authors declare no competing interests.

Received: 5 October 2023 Accepted: 17 July 2024
Published online: 02 August 2024

References

1. Airola A, Pyysalo S, Bjorne J, Pahikkala T, Ginter F, Salakoski T. All-paths graph kernel for protein-protein interaction extraction with evaluation of cross-corpus learning. BMC Bioinform. 2008;9(Suppl 11):S2.
2. Borgwardt K, Ghisu E, Llinares-López F, O'Bray L, Rieck B. Graph Kernels: state-of-the-art and future challenges. arXiv preprint. 2020. arXiv:2011.03854.
3. Borgwardt KM, Kriegel H-P. Shortest-path kernels on graphs, Fifth IEEE international conference on data mining (ICDM'05). IEEE. 2005; 8.
4. Bunescu R, Mooney R. A shortest path dependency kernel for relation extraction, Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing. 2005; 724–731.
5. Campos D, Matos S, Oliveira JL. Gimli: open source and high-performance biomedical name recognition. BMC Bioinform. 2013;14:54.
6. Chattopadhyay A, Lu TP. Gene-gene interaction: the curse of dimensionality. Ann Transl Med. 2019;7(24):813.
7. Chen SH, Sun J, Dimitrov L, Turner AR, Adams TS, Meyers DA, Chang BL, Zheng SL, Gronberg H, Xu J, Hsu FC. A support vector machine approach for detecting gene-gene interaction. Genet Epidemiol. 2008;32(2):152–67.
8. Cordell HJ. Detecting gene-gene interactions that underlie human diseases. Nat Rev Genet. 2009;10(6):392–404.
9. Craven M, Kumlien J. Constructing biological knowledge bases by extracting information from text sources. Proc Int Conf Intell Syst Mol Biol. 1999;1999:77–86.
10. Dijkstra EW. A note on two problems in connexion with graphs. Numer Math. 1959;1(1):269–71.
11. Faessler E, Hahn U, Schauble S. GePI: large-scale text mining, customized retrieval and flexible filtering of gene/protein interactions. Nucleic Acids Res. 2023;51(W1):W237–42.
12. Fang YH, Chiu YF. SVM-based generalized multifactor dimensionality reduction approaches for detecting gene-gene interactions in family studies. Genet Epidemiol. 2012;36(2):88–98.
13. Franceschini A, Szklarczyk D, Frankild S, Kuhn M, Simonovic M, Roth A, Lin J, Minguez P, Bork P, von Mering C, Jensen LJ. STRING v9.1: protein-protein interaction networks, with increased coverage and integration. Nucleic Acids Res. 2013;41:D808-815.
14. Gundem G, Lopez-Bigas N. Sample-level enrichment analysis unravels shared stress phenotypes among multiple cancer types. Genome Med. 2012;4(3):28.
15. Hagberg A, Swart P, Chult SD. Exploring network structure, dynamics, and function using NetworkX. Los Alamos: Los Alamos National Lab; 2008.
16. Hendrickx I, Kim SN, Kozareva Z, Nakov P, Séaghdha DO, Padó S, Pennacchiotti M, Romano L, Szpakowicz S. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. arXiv preprint. 2019. arXiv:1911.10422.
17. Hido S, Kashima H. A linear-time graph kernel, 2009 Ninth IEEE International Conference on Data Mining. IEEE. 2009; 179–188.
18. Huang M, Liu J, Zhu X. GeneTUKit: a software for document-level gene normalization. Bioinformatics. 2011;27(7):1032–3.
19. Jene-Sanz A, Varaljai R, Vilkova AV, Khramtsova GF, Khramtsov AI, Olopade OI, Lopez-Bigas N, Benevolenskaya EV. Expression of polycomb targets predicts breast cancer prognosis. Mol Cell Biol. 2013;33(19):3951–61.
20. Junge A, Jensen LJ. CoCoScore: context-aware co-occurrence scoring for text mining applications using distant supervision. Bioinformatics. 2020;36(1):264–71.
21. Kanehisa M, Sato Y. KEGG Mapper for inferring cellular functions from protein sequences. Protein Sci. 2020;29(1):28–35.
22. Karadeniz I, Hur J, He Y, Ozgur A. Literature mining and ontology based analysis of host-brucella gene-gene interaction network. Front Microbiol. 2015;6:1386.
23. Kim J, Kim JJ, Lee H. An analysis of disease-gene relationship from Medline abstracts by DigSee. Sci Rep. 2017;7:40154.
24. Koo CL, Liew MJ, Mohamad MS, Salleh AH. A review for detecting gene-gene interactions using machine learning methods in genetic epidemiology. Biomed Res Int. 2013;2013:432375.
25. Lamurias A, Clarke LA, Couto FM. Extracting microRNA-gene relations from biomedical literature using distant supervision. PLoS ONE. 2017;12(3):e0171929.
26. Lopez-Bigas N, De S, Teichmann SA. Functional protein divergence in the evolution of Homo sapiens. Genome Biol. 2008;9(2):R33.
27. Marneffe de M-C, Manning CD. Stanford typed dependencies manual. Stanford University, Stanford, CA, USA. Tech Rep. 2008; 338–345.
28. Mallory EK, Zhang C, Re C, Altman RB. Large-scale extraction of gene interactions from full-text literature using DeepDive. Bioinformatics. 2016;32(1):106–13.
29. Morgan AA, Hirschman L, Colosimo M, Yeh AS, Colombe JB. Gene name identification and normalization using a model organism database. J Biomed Inform. 2004;37(6):396–410.
30. Neumann M, Garnett R, Bauckhage C, Kersting K. Propagation kernels: efficient graph kernels from propagated information. Mach Learn. 2016;102(2):209–45.
31. Panyam NC, Verspoor K, Cohn T, Ramamohanarao K. Exploiting graph kernels for high performance biomedical relation extraction. J Biomed Semant. 2018;9(1):7.
32. Papanikolaou N, Pavlopoulos GA, Theodosiou T, Iliopoulos I. Protein-protein interaction predictions using text mining methods. Methods. 2015;74:47–53.
33. Previde P, Thomas B, Wong M, Mallory EK, Petkovic D, Altman RB, Kulkarni A. GeneDive: a gene interaction search and visualization tool to facilitate precision medicine. Pac Symp Biocomput. 2018;23:590–601.
34. Ritchie MD, Hahn LW, Moore JH. Power of multifactor dimensionality reduction for detecting gene-gene interactions in the presence of genotyping error, missing data, phenocopy, and genetic heterogeneity. Genet Epidemiol. 2003;24(2):150–7.
35. Shervashidze N, Schweitzer P, Van Leeuwen EJ, Mehlhorn K, Borgwardt KM. Weisfeiler-lehman graph kernels. J Mach Learn Res. 2011;12(9):2539.
36. Siglidis G, Nikolentzos G, Limnios S, Giatsidis C, Skianis K, Vazirgiannis M. GraKeL: a graph kernel library in python. J Mach Learn Res. 2020;21(54):1–5.
37. Singhal A, Simmons M, Lu Z. Text mining genotype-phenotype relationships from biomedical literature for database curation and precision medicine. PLoS Comput Biol. 2016;12(11):e1005017.
38. Su L, Meng X, Ma Q, Bai T, Liu G. LPRP: a gene-gene interaction network construction algorithm and its application in breast cancer data analysis. Interdiscip Sci. 2018;10(1):131–42.
39. Sugiyama M, Borgwardt K. Halting in random walk kernels. Adv Neural Inf Process Syst. 2015;28:1639–47.
40. Upstill-Goddard R, Eccles D, Fliege J, Collins A. Machine learning approaches for the discovery of gene-gene interactions in disease data. Brief Bioinform. 2013;14(2):251–60.
41. Wang S, Wu R, Lu J, Jiang Y, Huang T, Cai YD. Protein-protein interaction networks as miners of biological discovery. Proteomics. 2022;22(15–16):e2100190.
42. Wei CH, Harris BR, Kao HY, Lu Z. tmVar: a text mining approach for extracting sequence variants in biomedical literature. Bioinformatics. 2013;29(11):1433–9.
43. Wei CH, Kao HY. Cross-species gene normalization by species inference. BMC Bioinform. 2011;12(Suppl 8):S5.
44. Wei CH, Kao HY, Lu Z. SR4GN: a species recognition software tool for gene normalization. PLoS ONE. 2012;7(6):e38460.

45. Wei CH, Kao HY, Lu Z. PubTator: a web-based text mining tool for assisting biocuration. Nucleic Acids Res. 2013;41:W518-522.
46. Weisfeiler B, Leman A. The reduction of a graph to canonical form and the algebra which appears therein. NTI Ser. 1968;2(9):12–6.
47. Wiegers TC, Davis AP, Mattingly CJ. Collaborative biocuration–text-mining development task for document prioritization for curation. Database. 2012;2012:bas037.
48. Wimalagunasekara SS, Weeraman J, Tirimanne S, Fernando PC. Protein-protein interaction (PPI) network analysis reveals important hub proteins and sub-network modules for root development in rice (Oryza sativa). J Genet Eng Biotechnol. 2023;21(1):69.
49. Wong M, Previde P, Cole J, Thomas B, Laxmeshwar N, Mallory E, Lever J, Petkovic D, Altman RB, Kulkarni A. Search and visualization of gene-drug-disease interactions for pharmacogenomics and precision medicine research using GeneDive. J Biomed Inform. 2021;117:103732.
50. Yang L, Zhang YH, Huang F, Li Z, Huang T, Cai YD. Identification of protein-protein interaction associated functions based on gene ontology and KEGG pathway. Front Genet. 2022;13:1011659.
51. Yi N. Statistical analysis of genetic interactions. Genet Res (Camb). 2010;92(5–6):443–59.
52. Yuan F, Pan X, Chen L, Zhang YH, Huang T, Cai YD. Analysis of protein-protein functional associations by using gene ontology and KEGG pathway. Biomed Res Int. 2019;2019:4963289.
53. Zhang Y, Lin H, Yang Z, Li Y. Neighborhood hash graph kernel for protein-protein interaction extraction. J Biomed Inform. 2011;44(6):1086–92.
54. Zhang Y, Lin H, Yang Z, Wang J, Li Y. Hash subgraph pairwise kernel for protein-protein interaction extraction. IEEE/ACM Trans Comput Biol Bioinform. 2012;9(4):1190–202.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.