molecules

MDPI

Article

Synstable Fusion: A Network-Based Algorithm for Estimating Driver Genes in Fusion Structures

Mingzhe Xu [1,2,3], Zhongmeng Zhao [1,3], Xuanping Zhang [1,3,*], Aiqing Gao [1,3], Shuyan Wu [4] and Jiayin Wang [1,3,*]

1 Department of Computer Science and Technology, School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China; mingzhe.xu@hnuahe.edu.cn (M.X.); zmzhao@mail.xjtu.edu.cn (Z.Z.); algoxjtu@163.com (A.G.)
2 Department of Automation, College of Intelligent Manufacturing and Automation, Henan University of Animal Husbandry and Economy, Zhengzhou 450011, China
3 Shaanxi Engineering Research Center of Medical and Health Big Data, School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China
4 Department of Network Technology, College of Intelligent Manufacturing and Automation, Henan University of Animal Husbandry and Economy, Zhengzhou 450011, China; xxxwljys@126.com
* Correspondence: zxp@mail.xjtu.edu.cn (X.Z.); wangjiayin@mail.xjtu.edu.cn (J.W.); Tel.: +86-29-8266-8971 (J.W.)

Academic Editors: Xiangxiang Zeng, Alfonso Rodríguez-Patón and Quan Zou

Received: 25 June 2018; Accepted: 7 August 2018; Published: 16 August 2018

Abstract: Gene fusion structure is a class of common somatic mutational events in cancer genomes, which are often formed by chromosomal mutations. Identifying the driver gene(s) in a fusion structure is important for many downstream analyses and it contributes to clinical practices. Existing computational approaches have prioritized the importance of oncogenes by incorporating prior knowledge from gene networks. However, different methods sometimes suffer different weaknesses when handling gene fusion data due to multiple issues such as fusion gene representation, network integration, and the effectiveness of the evaluation algorithms. In this paper, Synstable Fusion (SYN), an algorithm for computationally evaluating the fusion genes, is proposed. This algorithm uses network-based strategy by incorporating gene networks as prior information, but estimates the driver genes according to the destructiveness hypothesis. This hypothesis balances the two popular evaluation strategies in the existing studies, thereby providing more comprehensive results. A machine learning framework is introduced to integrate multiple networks and further solve the conflicting results from different networks. In addition, a synchronous stability model is established to reduce the computational complexity of the evaluation algorithm. To evaluate the proposed algorithm, we conduct a series of experiments on both artificial and real datasets. The results demonstrate that the proposed algorithm performs well on different configurations and is robust when altering the internal parameter settings.

Keywords: gene fusion data; gene susceptibility prioritization; evaluating driver partner; gene networks

1. Introduction

Gene fusion is an important class of somatic mutational events in cancers [1]. A series of studies have shown that gene fusion structures, as well as the related genomic structural variations, are significantly associated with cancer susceptibilities across multiple cancer types [1–6]. With the development of sequencing technology, detecting gene fusion structures has become routine work in a number of computational pipelines for cancer sequencing data [7–9].

A fusion gene is typically formed by the interaction of two or more genes that are usually called partner genes. Normally, a fusion gene has a driver partner and one or more passenger partners, according to their roles in the evolution of tumor tissue [10]. The driver partner has a vital function in the carcinogenesis processes. Thus, identifying the driver partner is important for many downstream analyses and presents clinical implications. However, the throughput for validating driver genes is limited by current technology, which is both time consuming and expensive. A small number of the driver partners have demonstrated associations to cancer susceptibilities. Thus, computational approaches have been introduced to filter and prioritize the driver partner candidates, which facilitate and may further guide functional validations. To evaluate the importance of each partner in a gene fusion structure, gene networks are used in almost every existing approach, although different approaches vary in their use and application. A gene network is usually represented as a weighted graph, where each node in the graph denotes a gene, whereas each edge denotes a specific type of interaction between the two genes. Different types of approaches and interactions exist, including co-expression and co-localization networks [11–13], genetic interaction networks [14,15], pathway networks [16,17], physical interaction networks [18,19], shared protein domain networks [20], and predicted networks [21,22].

Along with the accumulation of gene network data, network-based approaches are faced with two major computational challenges. The first is determining how to incorporate knowledge from various types of networks. Multiple heterogeneous gene networks reflect different relationships. A common strategy involves establishing a virtual network by weighting the prior information from different networks. This is similar to the collapsing, or burden-test, strategy used in association studies [23] or the multi-source data-integration and decision-making process [24]. Benefiting from the amplification of the data signals via the newly collapsed network, the evaluation algorithms may be better for discovering potential associations and be more accurate in prioritizing the susceptibility genes [25–27]. Here, edge weight and graph structure are the two major evaluation strategies used to sort the important nodes (genes) through the collapsed network. The importance of edge weight is obvious, whereas the graph structure is considered by calculating the impact of each node based on the network, such as node degree [28] and node betweenness [29]. Node degree is a local topology strategy that only computes the weights on the edges that directly connect to the node. Node betweenness provides a global view by presenting the connectivity influence of nodes on the entire network. The existing approaches, however, are usually sensitive to the incorporation of the networks. When a neural network is collapsed into a disease-associated network, it may excessively dilute the data signal [23]. For example, in the multi-layer design of neural networks [24,30], multiple disease-associated network data are merged into a single output signal. Most of the data being processed within neural networks are eliminated by the weights of the input layer and the activation function of neurons in hidden layers [30].

The second major computational challenge is addressing the conflicting results from different networks. Different from the point mutation or indel calls, a gene fusion structure consists of two or more partner genes. Gene networks do not contain any "combined" nodes corresponding to a fusion gene. Thus, in many cases, the evaluation algorithms may provide conflicting results on the same virtual network. To solve the conflicts, after extensive experimental verifications [31–34], Wu et al. [34] provided the hypothesis that "if a fusion gene plays an important role in tumor formation, then the partner genes should be an important node in the gene network". This hypothesis, called "network fusion centrality", is based on many previous research works, which concluded that all partner genes of the carcinogenic fusion gene usually have higher network centrality, and suggested that oncogenes prefer hub nodes in the network. The network fusion centrality hypothesis allows the algorithms to merge the nodes that correspond to the partner genes into a burden node representing the fusion gene [34]. In this case, the importance of a gene fusion structure is the accumulation of the importance of the previous partner nodes. However, some of the information between the nodes, which may be lost due to the overlapped edges of merged partner gene nodes, is often ignored.

Multiple approaches are available for prioritizing partner genes, among which network fusion centrality (FC) strategy is popular [28,29,34], as it is able to process gene fusion data better than other existing approaches. In this strategy, the gene networks are obtained as prior knowledge, each of which contains a set of genes. Note that, each node of these networks represents a single gene, and each edge denotes a specific type of interaction between the two genes. As none of the nodes represent a fusion gene or a gene fusion structure, the fusion gene nodes are constructed by merging the corresponding partner genes. To achieve this, the merging step first maps the partner genes to the entire gene network, and then each partner inherits the functions of the original gene on the network to evaluate the potential influence.

Two evaluation strategies for measuring the importance of a node are widely used: node degree [28] criterion and node betweenness [29] criterion. In the node degree algorithm, the degree of node i is calculated with $K(i) = \frac{\sum_{j \in G} a_{ij}}{N-1}$, where N represents the number of nodes and G represents the set of nodes. For unweighted networks, $a_{ij} \in (0, 1)$, where 0 indicates that no edge exists between node i and node j, and 1 indicates that an edge exists. For weighted networks, a_{ij} denotes the edge weight between nodes, where $K(i)$ represents the weighted degree of a node. The degree of the node represents the direct connection state between the node and other nodes. The importance of the node is expressed by the number of directly connected nodes. This method evaluates the significance of a node based on how well the node is directly connected to other nodes in the network topology. The advantage of this method is that the calculation is simple and the algorithm's time complexity is $O(N^2)$. The disadvantage is that only the neighbors of the node are considered, and only the local importance of the nodes in the network is calculated. For nodes in different positions in a complex network, the node importance caused by various topologies is not considered.

Node betweenness is a parameter used by Freeman [29] to measure social status of individuals in their research on social networks. The betweenness of node k is defined as the number of shortest paths between any two nodes passing through node k. The betweenness centrality $B(k)$ of node k is defined as $B(k) = \frac{g(k)}{g}$, where g is the number of shortest paths between each pair of nodes, and $g(k)$ is the number of the shortest paths via node k. The larger the value of node betweenness, the greater the role played by the node in the connectivity between other nodes in the network. That is, the greater the influence of the node on the network connectivity, the more important the node to the entire network. The node betweenness mainly considers the impact of nodes on the connectivity between other nodes in the network. The advantage is that the global importance of a node is explained by the impact of the node on the shortest paths between nodes in the entire network. The disadvantage is that the interaction between directly connected nodes is ignored, and the method is highly complex because it is time consuming to find the shortest path between all nodes.

The algorithm based on fusion centrality degree (DEG) [28,34] uses the degree of fusion node as the evaluation measurement, whereas the algorithm based on fusion centrality betweenness (BET) [29,34] evaluates the fusion nodes based on the betweenness. However, these criteria have been further argued to have their own preferences; thus, more comprehensive strategies are suggested. Other than the degree and betweenness, graph stability is another important measurement in graph theory to describe destructiveness of a network. The graph stability state is gradually approximated if all of weights of the edges satisfy a necessary condition [35]. The necessary condition is determined by the size of the network, average connectivity among the nodes, and a coupling coefficient that relies on graph topology. Existing studies have proposed multiple synchronous stability criteria for various graph topologies [35]. For example, many networks have a semi-ring $2K$ adjacent sub-structure, which enables existing conclusions on synchronous stability criteria, widely extensible to more complicated gene network topologies. Specifically, when $k = 1$, the graph degenerates to a ring structure, whereas if $k = n/2$, the graph is a fully connected graph. Once the synchronous stability criteria are locked in the evaluation algorithm, the calculation complexity for the edge weight condition considerably decreases compared to the betweenness calculation.

To overcome the disadvantages of the current methods, and to evaluate the cancer susceptibility created by a fusion gene based on the synchronous stability method, an algorithm named Synstable Fusion is proposed in this paper. Synchronous stability means that the coupled network is synchronously stable if the internal coupling matrix and the network coupling matrix satisfy certain conditions [35]. The proposed algorithm calculates the importance of genes in the gene network according to the "destructiveness equals to importance" hypothesis [28,34,36], which states that the importance of a node in a connected graph is identical to the destructiveness of deleting the node, and evaluates the corresponding fusion genes through the importance of partner gene nodes. The Synstable Fusion algorithm, which is based on synchronous stability, evaluates the importance of the fusion gene according to the whether or not the gene network achieves a synchronously stable state. When a weighted network falls into a synchronously stable state, the network ignores the noise and insignificant information while retaining the important node edges and network structure as much as possible, thereby reducing the computational complexity when evaluating the overall impact of the node on the network. The destructiveness of deleting the node is measured by using the network difference criterion, which reflects the importance of the gene nodes. This approach not only considers the local importance of the node, but also measures the influence of the node on the overall network structure, so the gene node's importance can be accurately calculated. The performance of our algorithm is tested and compared to the DEG and BET algorithms in a series of experiments. The experimental results demonstrate that the Synstable Fusion algorithm is able to effectively evaluate cancer fusion genes and performs better than the existing method.

2. Results

In order to test and verify the effectiveness of the proposed algorithm, named Synstable Fusion, we applied the algorithm to a widely used whole-gene network [36] obtained by 17 heterogeneous data to evaluate the importance of the fusion genes represented by the nodes. This gene network was obtained from Wu et al. [36], and the edge weights in the network indicate the tendency of the two genes to be joined to work together in one pathway. This network not only represents direct interactions between genes, but also includes functional interactions in a broader sense and has been used in many pathological and therapeutic studies related to cancer genes [37–41]. The 40,230 genes included in the entire gene network are provided in the Supplementary Materials. In order to reflect as much key and useful information as possible, the network has to be further processed to retain reliable inter-gene interactions. In the experiments, we used the "network fusion centrality" hypothesis, which was also used in many subsequent studies [42–46].

2.1. Experimental Data

The experimental data were selected based on the above studies [34,36]. A gene whose mutation is associate to a disease is called a susceptible gene. We followed the hypothesis that fusion genes formed by the interaction of susceptible cancer genes have relatively high significance, since susceptible cancer genes are important for the production of cancer [31–34]. We extracted 699 professionally curated human oncogenes from the Cancer Gene Census (CGC) [47] project as the susceptible cancer fusion genes, from which cancer may result due to their mutations. The CGC project collects and validates all published cancer-related genetic mutation studies by professionals in the field, collating them into a database with filtering criteria, and updates and maintains the data. Oncogenic mutations include both single-gene mutations (amplification, insertion, deletion, etc.) and translocations (fusions). Thus, this oncogene list also contains all possible partner genes of known oncogenic fusion genes until the date (December 2017) we obtained the list (Supplementary Table S1).

In the test data, it is assumed that N_f represents the number of total fusion genes, N_i is the number of susceptible fusion genes, and N_o is non-susceptible fusion genes. So, $N_f = N_i + N_o$. Two partner genes form a fusion gene, thus the number of partner genes in the dataset is $2N_f = 2N_i + 2N_o$. To generate the dataset, we randomly selected $2N_i$ susceptible partner genes from

the known susceptible cancer genes [47], then paired them to create N_i susceptible fusion genes. For non-susceptible fusion genes, we randomly picked $2N_o$ common partner genes from the whole-gene network [36] and selected pairs to create N_o ordinary fusion genes. Here, we simply used the random function in the programming language's built-in library to implement random sampling without replacement process, and reset the random seed before each random process to ensure irregularity. N_i important fusion genes were assembled from paired samples of $2N_i$ oncogenes by random sampling without replacement. The same random sampling method was applied to $2N_o$ common genes to extract pairs of genes into N_o common fusion genes. The possibility of repeating samples inside the N_i and N_o datasets was avoided because the non-return sampling method was adopted. Since the whole gene network also contained 699 oncogenes for formation of susceptible fusion genes, and the selection process of N_i and N_o was independent of each other, overlaps between the important fusion genes (N_i) and the ordinary fusion genes (N_o) of one dataset occurred. Once this happened, we re-selected $2N_o$ common genes and randomly generated N_o fusion genes until no duplication was present between susceptible fusions and ordinary fusions. We prepared two N_i configurations and three N_f ($N_f = N_i + N_o$) configurations for the experiment, and 20 sets of random data were selected for each $N_i + N_o$ configuration, so there were $2 \times 3 \times 20 = 120$ sets of data in total. Every set of experimental data was created accordingly. Real known oncogenic fusion genes can be created using this procedure. Three expert-curated carcinogenic fusion genes, *EWSR1-FEV*, *HMGA2-LPP*, and *EWSR1-ETV4*, were identified from the datasets. All were assessed at high importance rankings by our evaluation algorithm. The results of respective datasets are provided in Supplementary Table S2.

2.2. Experimental Results

The effects of the SYN algorithm are illustrated using three criteria: (1) distribution curve of susceptible fusion gene; (2) recognition rate; and (3) receiver operating characteristic curve. The test applied various values of N_f and N_i. The effectiveness of the SYN algorithm was validated by the comparison with the DEG and BET algorithms. Different experiment scenarios were created based on various N_f and N_i values. For $N_f \in \{150, 200, 250\}$ and $N_i \in \{15, 25\}$, a total of six parameter configurations were generated. For each configuration, 20 sets of data were randomly generated. Our algorithm and the other two algorithms were applied to each set to separately calculate the importance and then sort the fusion genes according to these values. The results of the different configurations and algorithms are statistically summarized and the average data calculated from the 20 results of each case are demonstrated in the following subsections.

2.2.1. Distribution Curve of Susceptible Fusion Gene

The susceptible fusion genes were divided into 10 intervals I_i (i = 1, 2, . . . , 10), where $I_i = \left(\frac{i-1}{10}N_f, \frac{i}{10}N_f\right]$. For each dataset, all calculated fusion gene significance was sorted in descending order, and then separated into 10 ranking intervals. The number of susceptible fusion genes that fell under each interval were counted. The results showed the effect of SYN, DEG, and BET algorithms in six cases of $N_f \in \{150, 200, 250\}$ and $N_i \in \{15, 25\}$. Figure 1 shows the average distribution curves of the susceptible fusion genes identified by the three algorithms.

From the distribution curves of the susceptible fusion genes, SYN was able to find most of the significant fusion genes from the first two intervals. In $N_i = 15$ cases, the mean number of susceptible fusion genes in the top two intervals was 13.367, and this number was 21.4 in $N_i = 25$ situations. There were approximately zero susceptible fusion genes in the lowest five ranges. Therefore, we summarize the average number of susceptible fusions in the top 20% ranked fusion genes in various cases in Figure 2.

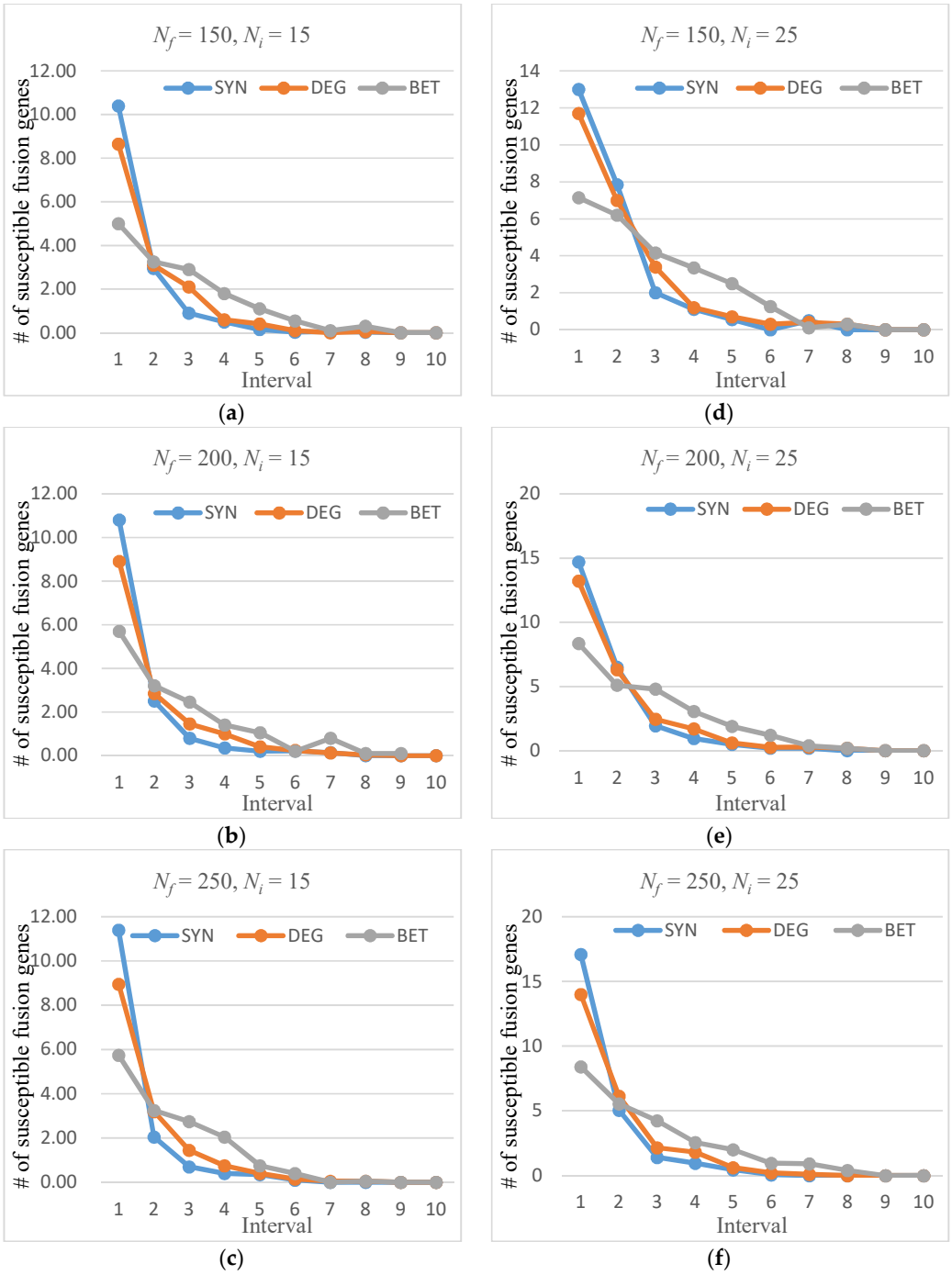
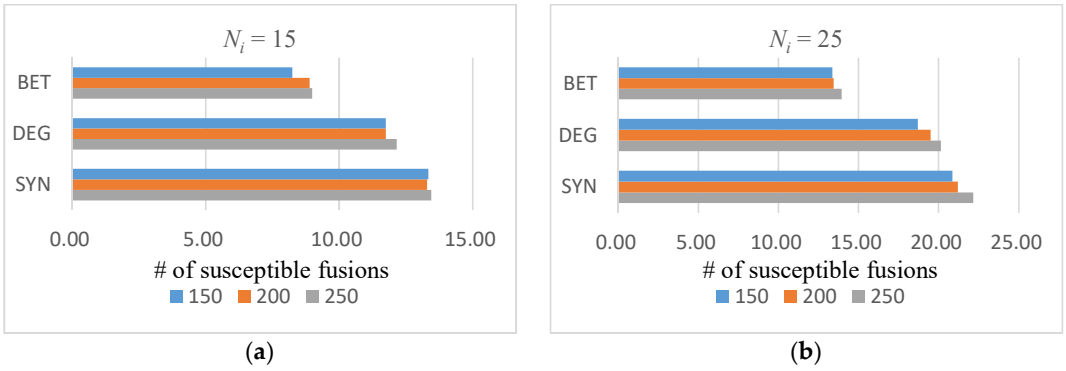


Figure 1. Average distribution curve of susceptible fusion genes: (**a**) $N_f = 150$, $N_i = 15$; (**b**) $N_f = 200$, $N_i = 15$; (**c**) $N_f = 250$, $N_i = 15$; (**d**) $N_f = 150$, $N_i = 25$; (**e**) $N_f = 200$, $N_i = 25$; and (**f**) $N_f = 250$, $N_i = 25$.



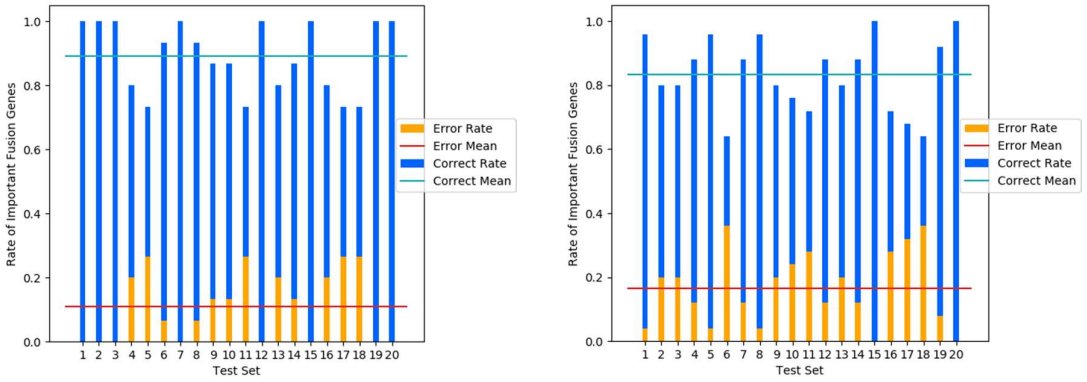
(**a**) (**b**)

Figure 2. Average number of susceptible fusions in the top 20%. Different colors indicate different cases of total fusion gene amounts: (**a**) $N_i = 15$ and (**b**) $N_i = 25$.

From Figure 2, the average results of the SYN algorithm always outperform the results obtained with the DEG and BET methods in all situations. The largest difference occurred when $N_i = 25$ and $N_f = 250$: the average number of susceptible fusions found by the SYN algorithm was 58.8% more than the BET algorithm. The smallest gap occurred in comparing with the DEG algorithm when $N_i = 15$ and $N_f = 250$, as the difference percentage was 10.7%. From these results in Figure 2, we found that as the total number of fusion genes increased notably (by one-third or one-quarter), the amount of susceptible fusions within the top 20% area did not increase considerably, and the ratios were lower than the increasing rates of total fusion genes. We will discuss possible reasons for this result later in the Discussion section.

2.2.2. Recognition Rate

In order to illustrate the effectiveness of the proposed algorithm, the recognition rate of the susceptible fusion gene was adopted. The recognition rate represents the ratio of susceptible fusion genes located in a statistical interval to the total susceptible fusion genes. The recognition rate P is presented as $P(i) = \frac{f(R(i))}{N_i}$, where $R(i)$ denotes the i^{th} statistical interval, $R(i) = \left[1, \frac{i}{10}N_f\right]$, $i \in \{1, 2, \ldots, 10\}$, and $f(R(i))$ indicates the number of susceptible fusion genes being found in the ith interval. We randomly selected 120 sets, and generated mixed experimental data in six cases ($N_f \in \{150, 200, 250\}$, $N_i \in \{15, 25\}$). As an illustrative case, Figure 3 demonstrates the $p(2)$ value of every experimental result.



(**a**) $N_f = 150,\ N_i = 15$ (**d**) $N_f = 150,\ N_i = 25$

Figure 3. *Cont.*

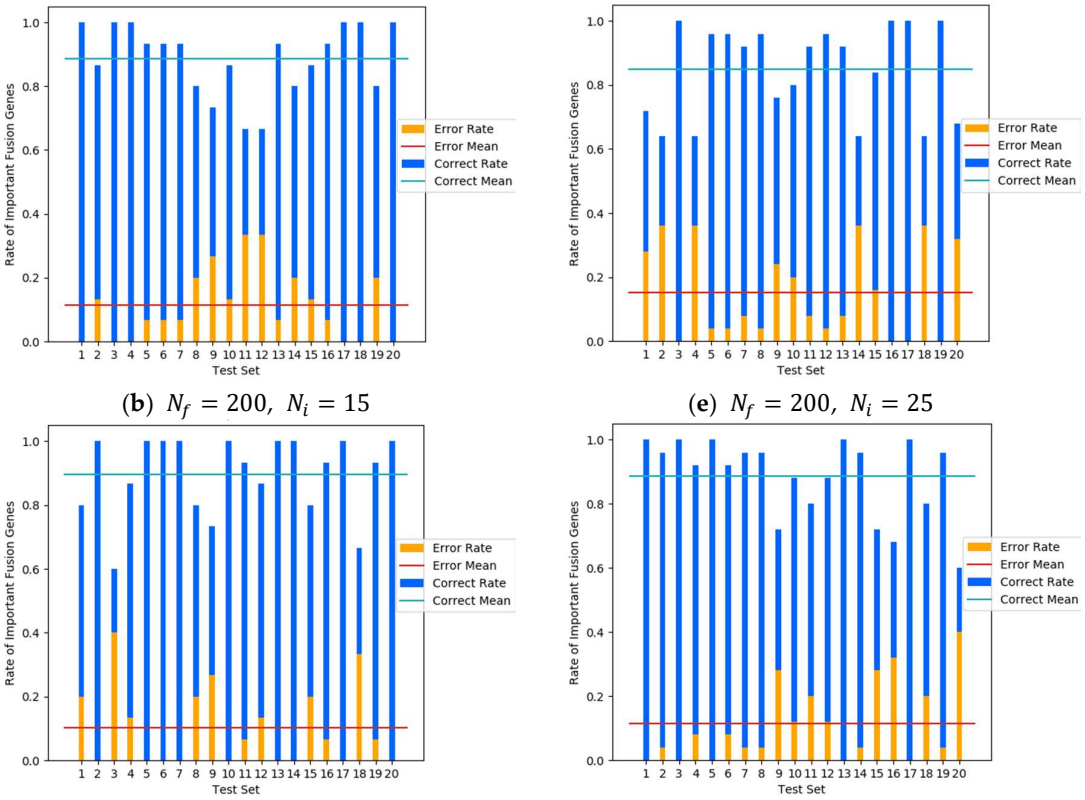


Figure 3. Recognition rates (correct rate) $p(2)$ (blue bars) of susceptible fusion genes in the top 20% of ranked fusion genes in each experimental result. The orange bar represents the rate of important fusion genes which is not included in this interval (error rate). (**a**) $N_f = 150$, $N_i = 15$; (**b**) $N_f = 200$, $N_i = 15$; (**c**) $N_f = 250$, $N_i = 15$; (**d**) $N_f = 150$, $N_i = 25$; (**e**) $N_f = 200$, $N_i = 25$; and (**f**) $N_f = 250$, $N_i = 25$.

The summarized average results are exhibited using radar panels, where vertices indicate the statistical intervals of the susceptible fusions, and axes indicate that recognition rate, which gradually increased outward. Because almost all susceptible fusion genes were included in the top 50% of ranked result, only the first five intervals are shown in the figures.

Figure 4 shows the $N_i = 15$. $p(1)$ results of the SYN algorithm. The recognition rate was about 70%, whereas those for the same interval obtained by the other two algorithms were less than 60%. The $p(2)$ value of the SYN algorithm was around 90%, which means approximately 90% of the susceptible fusion genes can be found using the SYN algorithm from its top 20% sorted results. As the range continuously increased, the $p(i)$ value increased as well. The differences among algorithms continually decreased and the recognition rates of all algorithms gradually approached 100%.

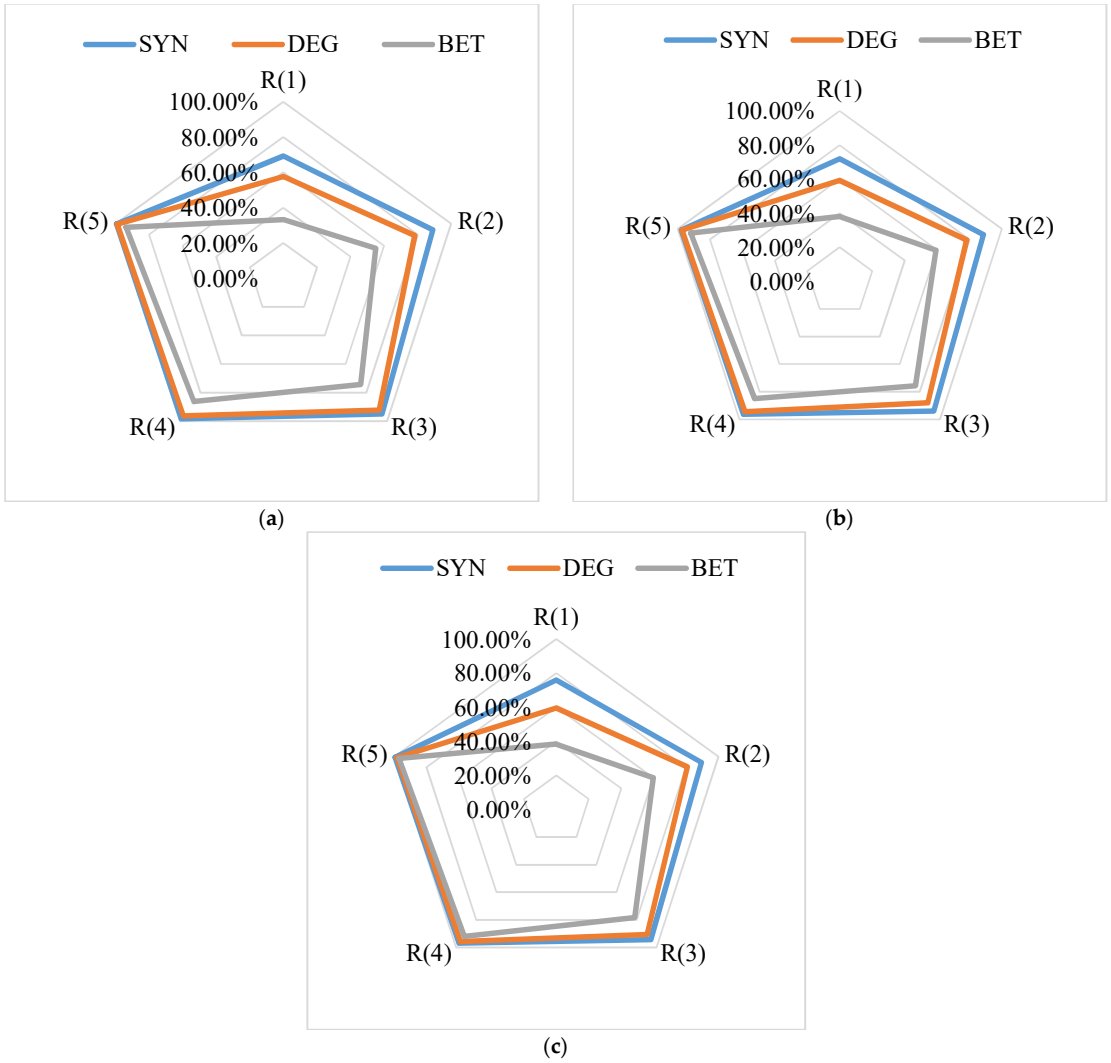


Figure 4. Average recognition rates of susceptible fusion gene in each interval when $N_i = 15$: (**a**) $N_f = 150$; (**b**) $N_f = 200$; and (**c**) $N_f = 250$.

Figure 5 highlights the statistical average results when $N_i = 25$. Compared with the case when $N_i = 15$, the overall recognition rate of the interval $R(1)$ decreased significantly, which was mainly because the total number of fusion genes in $R(1)$ was less than or equal to the number of pathogenic fusion genes in test samples ($N_i = 25$). The corresponding proportion of pathogenic fusion genes was relatively lower. Other factors also affected the experimental results. We discuss the possible causes in the Discussion section. The $p(2)$ value of the SYN algorithm was around 85%, whereas the recognition rates for the same interval obtained by the two other control algorithms were both less than 80%. The $p(3)$ value of the SYN algorithm was higher than 90%, whereas the values obtained by the two control algorithms were less than 90%. Figures 4 and 5 clearly illustrate that the recognition rate of the SYN algorithm in all situations was higher than those of the DEG and BET methods.

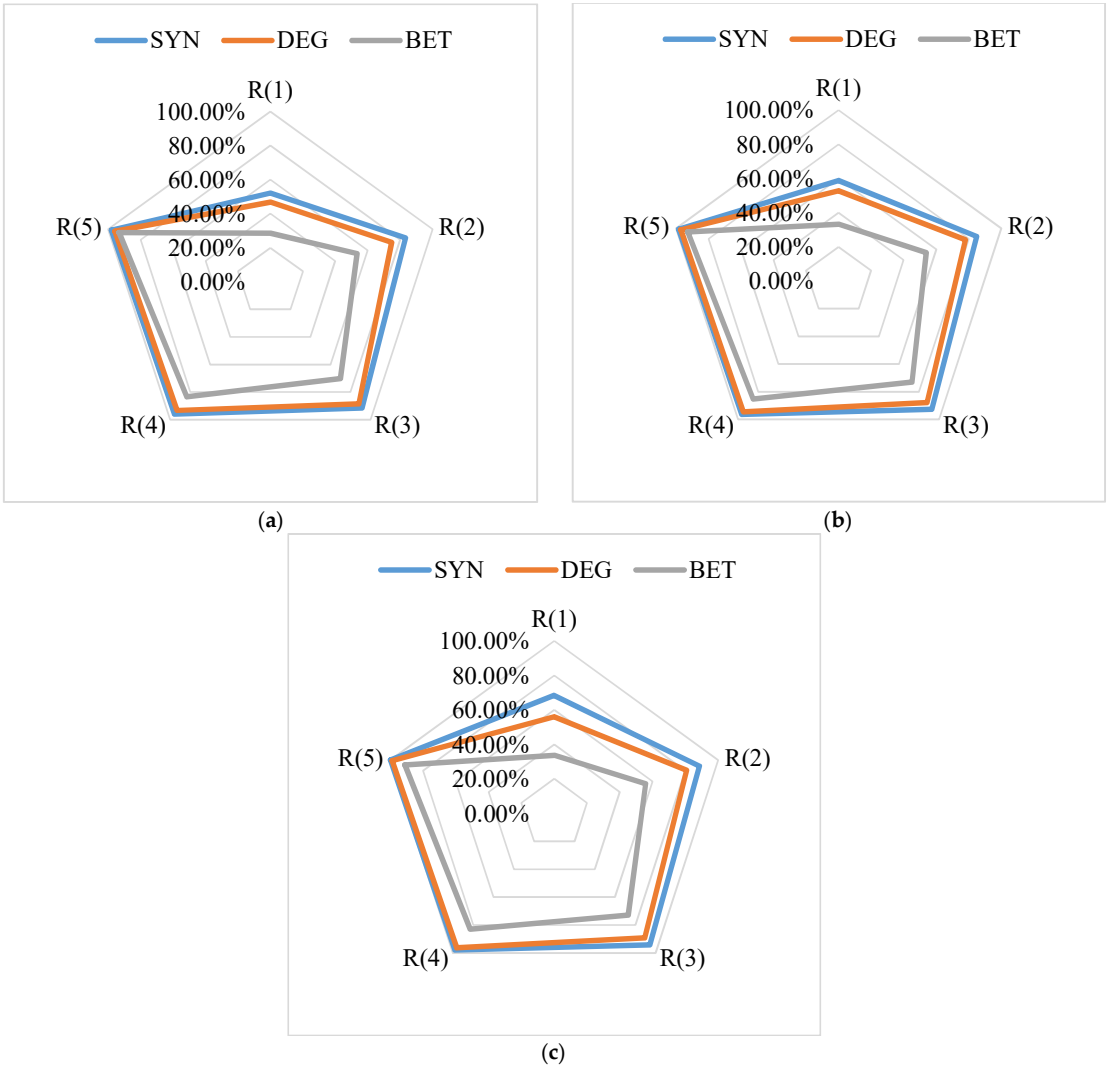


Figure 5. Average recognition rates of susceptible fusion gene in each interval when $N_i = 25$: (**a**) $N_f = 150$; (**b**) $N_f = 200$; and (**c**) $N_f = 250$.

2.2.3. Receiver Operating Characteristic Curve

By adding a classification boundary to the results of the algorithms, the original algorithm can be changed into a binary classification algorithm. Fusion genes above the classification limit can be classified as cancer pathogen fusion genes, and vice versa as normal fusion genes. As such, we calculated the algorithm's receiver operating characteristic curve (ROC). Figure 6 shows the ROC curves of the three algorithms in all six cases and the area under curve (AUC) values for each curve, where the Y-axis is the true positive (TP) rate and the X-axis is the false positive (FP) rate.

From the ROC results, the best classification performance occurred at N_f=150 and $N_i = 15$, where the AUC value was around 0.945. The situation with the smallest AUC score was $N_f = 200$ and $N_i = 15$, which had a value around 0.93. The overall performance of the proposed algorithm remained high.

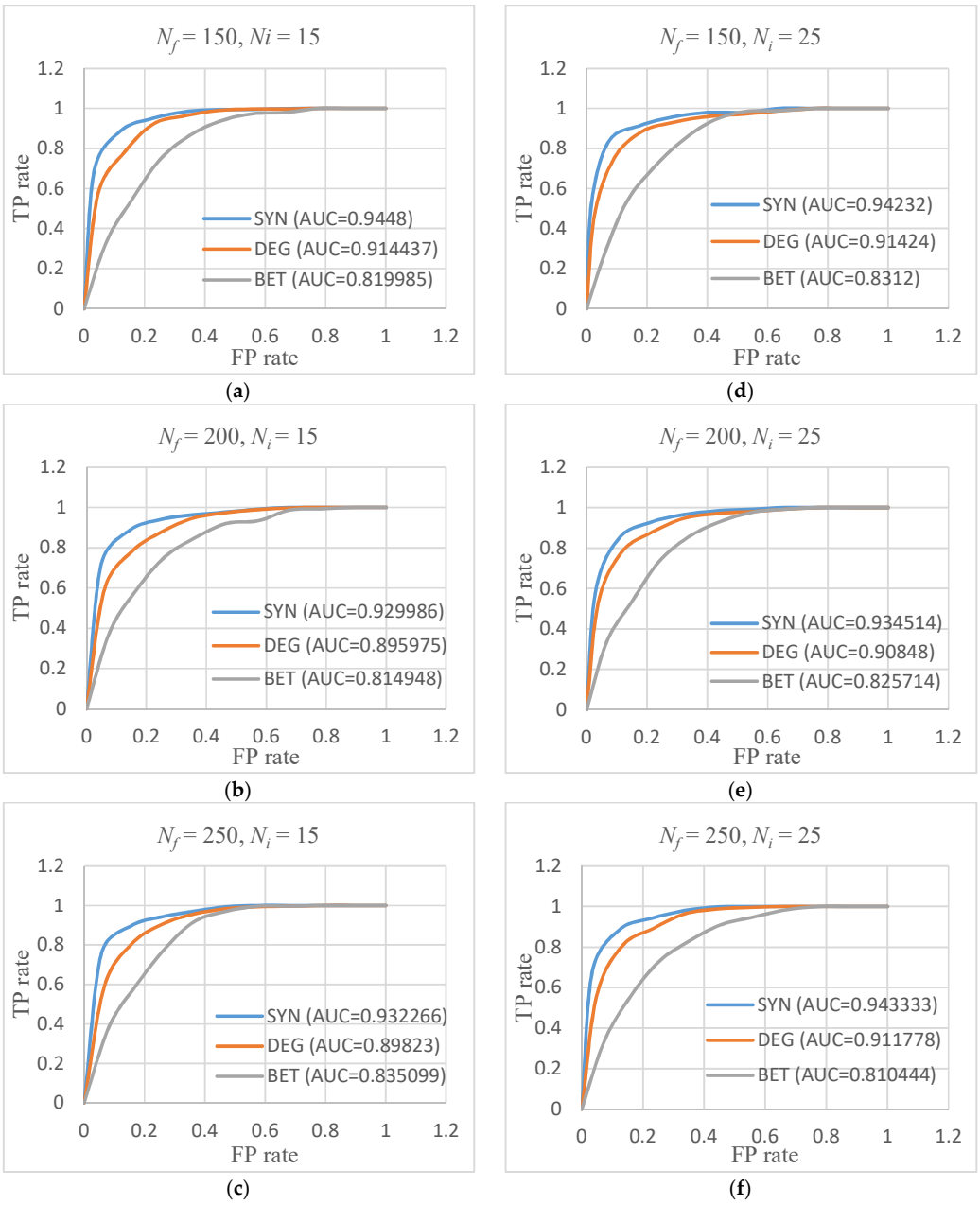


Figure 6. Receiver operating characteristic (ROC) curves of the three algorithms: (**a**) $N_f = 150$, $N_i = 15$; (**b**) $N_f = 200$, $N_i = 15$; (**c**) $N_f = 250$, $N_i = 15$; (**d**) $N_f = 150$, $N_i = 25$; (**e**) $N_f = 200$, $N_i = 25$; and (**f**) $N_f = 250$, $N_i = 25$. FP—false positive; TP—true positive.

3. Discussion

The experimental results clearly demonstrate that the proposed Synstable Fusion (SYN) algorithm performs better when calculating the importance of cancer-causing fusion genes in gene networks. More susceptible fusion genes were included in the top portion of the descending-sorted result, which means that possible oncogenic fusion genes have a greater tendency to be evaluated with higher importance values when using the SYN algorithm. As an example, three known oncogenic fusion genes found in the experimental results obtained by our algorithm received high importance rankings. The three fusion genes, EWSR1-FEV, HMGA2-LPP, and EWSR1-ETV4, were ranked first, tenth, and second in their respective datasets. Specific experimental datasets, importance calculation scores, and potential carcinogenic rankings are provided in Supplementary Table S2.

We found two phenomena worth noting. The first is that the number of cancer pathogenic fusion genes identified in the first 20% of the ranking results did not increase significantly as the total number of samples included in the test dataset increased. At $N_i = 15$, the discrepancy among the maximum and minimum numbers of pathogenic fusion genes in the first 20% of the results of SYN algorithm was only 0.15 in three cases, and this difference only increased to 1.3 at $N_i = 25$. The second phenomena is that the overall recognition effect slightly decreased when the total number of pathogenic fusion genes in the sample was high. For example, when $N_f = 250$, the five-interval average recognition rate of the cancer-causing fusion gene of the SYN algorithm was 91.27%; when $N_i = 15$, this value was 89.8%, and when $N_i = 25$, a decrease of about 1.5% was observed. In the following, we discuss the possible causes of these two phenomena and explain why the results of the proposed algorithm are better than those of the other two algorithms.

The first case phenomenon occurred when the number of identified pathogenic fusion genes did not increase with the total fusion gene number in the sample. This may have occurred because the calculation scores of most disease-causing fusion genes were high but the scores of a fixed fraction of the proportion of susceptible fusion genes were lower. This is because the algorithms' results are based on the genomics inference network derived by the classification algorithm of machine learning, and the result generated by classification algorithms must be partially consistent with the expected errors.

In the experimental dataset, the number of susceptible fusion genes was high whereas the recognition effect was slightly lower, possibly because the increase in the number of pathogenic fusion genes in the samples led to an increase in the occurrence probability of susceptible cancer fusion genes with low importance scores. The distribution of the number of pathogenic fusion genes in the first 20% results can provide support for this explanation. When $N_i = 15$, the recognition distribution of each algorithm (Figure 2a) was almost unchanged, and the number of high-importance pathogenic fusion genes remained unchanged at a high rate. The number of identifications at $N_i = 25$ (Figure 2b) slightly increased because some of the disease-causing fusion genes with slightly lower scores appeared in the test dataset. These genes were gradually identified as the range of recognition intervals increased.

From the experimental results, the performance of the proposed algorithm is better than that of the DEG and BET algorithms under various parameter settings with experimental data. The proposed algorithm uses more comprehensive information contained in the gene network to calculate the importance of nodes. When evaluating the importance of nodes, BET algorithm only considers the influence of the nodes on the network topology, whereas the DEG algorithm only considers the influence between the node and its directly related parts of the network. However, in our algorithm, the "destructiveness equals to importance" hypothesis is applied, which not only considers the degree of the node, but also the impact of deleting the node on the network topology. This is equivalent to a certain degree of the incorporation of the first two algorithms. Therefore, SYN outperforms the DEG and BET algorithms.

4. Materials and Methods

The Synstable Fusion algorithm is based on the synchronous stability method, which evaluates the node importance according to the influence on stability of gene network when a node is removed. Wu et al. [36] used a Relevance Vector Machine (RVM)-based [48] ensemble-learning model to construct a whole gene network. This model integrates 17 heterogeneous genomic data and proteomics data [36]. We used this model mainly because it incorporates many different kinds of data, and simultaneously better handles the problem of missing attribute values among heterogeneous data [49] and outputs probabilistic results. Weighted edges existed between paired nodes in the entire gene network of the human genome. The weight represents the probability of interactive works between two genes, not only reflecting the direct interactions between genes, such as activation, inhibition, binding, and dissociation, but also other broader relationships among genes, for example, the likelihood of genes working on the same or similar biological pathways. In order to evaluate the influence of a node on stability, the synchronously stable networks were identified from the original gene network

and the network of deleting a node, and then the difference between these two synchronously stable networks was calculated. Based on the node influence on network stability, the cancer fusion gene was evaluated. In this section, the design of the proposed algorithm is described in detail.

4.1. Synchronous Stability Method

In order to identify the synchronously stable network from a gene network, the synchronous stability method was required to ensure the relative stability of the gene network. A network is considered to be in synchronously stable state when it satisfies a certain condition.

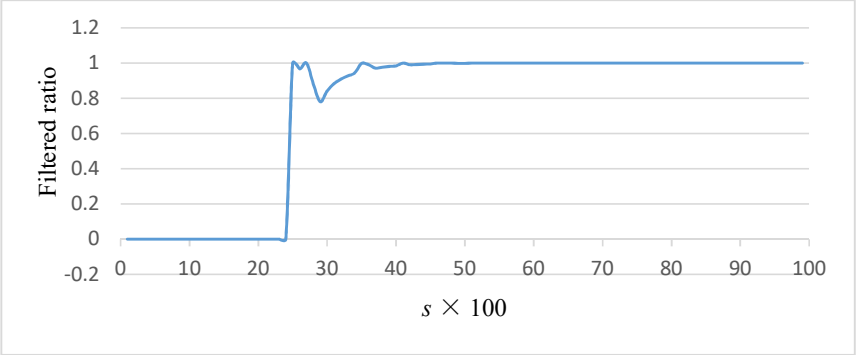
4.1.1. Synchronous Stability Condition

For the connected graph with ring of $2K$ adjacent nodes, the condition of synchronous stability is presented [35] as:

$$w > \varepsilon = \frac{a}{n}\left(\frac{n}{2K}\right)^3\left(1+\frac{65}{4}\frac{K}{n}\right) \quad (1)$$

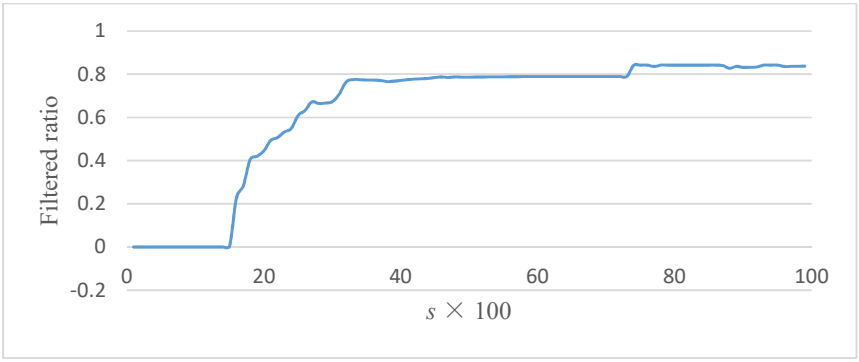
where w denotes the edge weight, ε represents the lowest limit of the w, a is an important parameter indicating the coupling state of network, n denotes the node amount of the network, and K indicates the number of half neighbors. Parameter a is called the coupling parameter that describes the coupling characteristic of the network. The value of a is determined by analyzing the adjacency matrix of graph. A previous study [50] indicated that λ_2 is the algebraic connectivity of the connected graph and a $< \lambda_2$. By inducing the Laplace matrix, we obtained a series of eigenvalues that satisfy $0 = \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$. The algebraic connectivity λ_2 is one of the eigenvalues and it was the minimum nonzero-eigenvalue. λ_2 denotes the synchronous ability of the connected graph. Thus, $0 < a < \lambda_2$ and then we sequentially chose the fittest a value from this range based on some system analysis.

In order to find the most suitable value a, let $a = s\lambda_2$ and s = [0.01, 0.02, ..., 0.99]. For each s value, we calculated the proportion of the lost information filtered by the synchronously steady state of a given gene network. In the experimental gene data, 20 gene networks were randomly selected and generated. Figure 7 shows the result of one set of data, the average result of 20 sets of data, and the gradient of the average result.

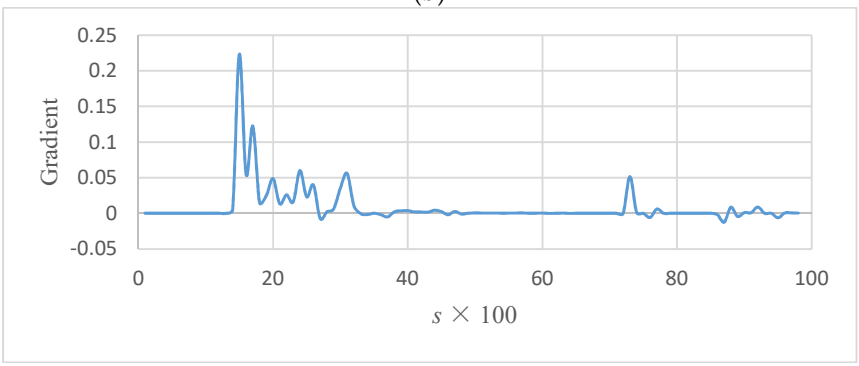


(**a**)

Figure 7. *Cont.*



(**b**)



(**c**)

Figure 7. The results by various *s* (the product factor of coupling state parameter) values: (**a**) result of one set of data; (**b**) average result of 20 sets of data; and (**c**) Gradient of average result.

We tried to find a suitable *s* value for most experimental data to filter out most noise and insignificant information while retaining key information in gene network. From Figure 6a,b, we found that depression points always occurred in the proportion of filtered data in all 20 experimental results. Through analyzing the gradient of the average data, we found the depression point where the gradient was first close to zero. From Figure 6c, point 0.28 satisfies the requirement. So $a = 0.28\lambda_2$, which is inserted into Equation (1):

$$w > \varepsilon = \frac{0.28\lambda_2}{n}\left(\frac{n}{2K}\right)^3\left(1 + \frac{65}{4}\frac{K}{n}\right). \tag{2}$$

Our research considered two situations of the connected graphs: the gene networks have a fully connected topology, and the gene networks do not have a fully connected topology. The synchronous stability condition of fully connected networks can be obtained by letting $n = 2K$. Therefore, Equation (2) becomes:

$$w > \varepsilon = 2.555\frac{\lambda_2}{n}. \tag{3}$$

If the topology of graph is not fully connected, its maximum fully connected subgraph can be found. Let m denote half of the number of nodes in this subgraph. For the connected graph with a maximum ring of $2K$ adjacent nodes, the fully connected subgraph is a ring of $2m$ adjacent nodes, so we obtain $2m < 2K$. The value of ε is increased by replacing k with m:

$$\mathrm{w} > \varepsilon = \frac{0.28\lambda_2}{n}\left(\frac{n}{2m}\right)^3\left(1 + \frac{65}{4}\frac{m}{n}\right). \tag{4}$$

The edge weight limit ε calculated by Equation (4) is greater than the lowest limit of the synchronously stable condition. Therefore, the new limit can also be used as the judging condition for synchronous stability.

4.1.2. Identification of Synchronously Stable Network

From Equations (3) and (4), we obtained the lower edge weight limit for every gene network of various topologies. The network is in a synchronously steady state if $\forall w_{ij} > \varepsilon$, where w_{ij} denotes the edge weight between node *i* and *j*. Otherwise, if $\exists w_{ij} < \varepsilon$, it is in a non-synchronously steady state and needs further processing to achieve synchronous stability. Different procedures were assigned based on whether or not the gene network was fully connected. All edges with a weight less than the lower limit were deleted if the network was fully connected. If the connected subgraph was a non-fully connected graph, the hanging nodes were detected. If there were hanging nodes, the hanging nodes were deleted. If no hanging nodes existed in the connected subgraph, then the edges with the weight less than the low limit were deleted. This procedure was iterated until all connected subgraphs were in the synchronously steady state. Figure 8 shows the flowchart of the identification of the synchronously stable network.

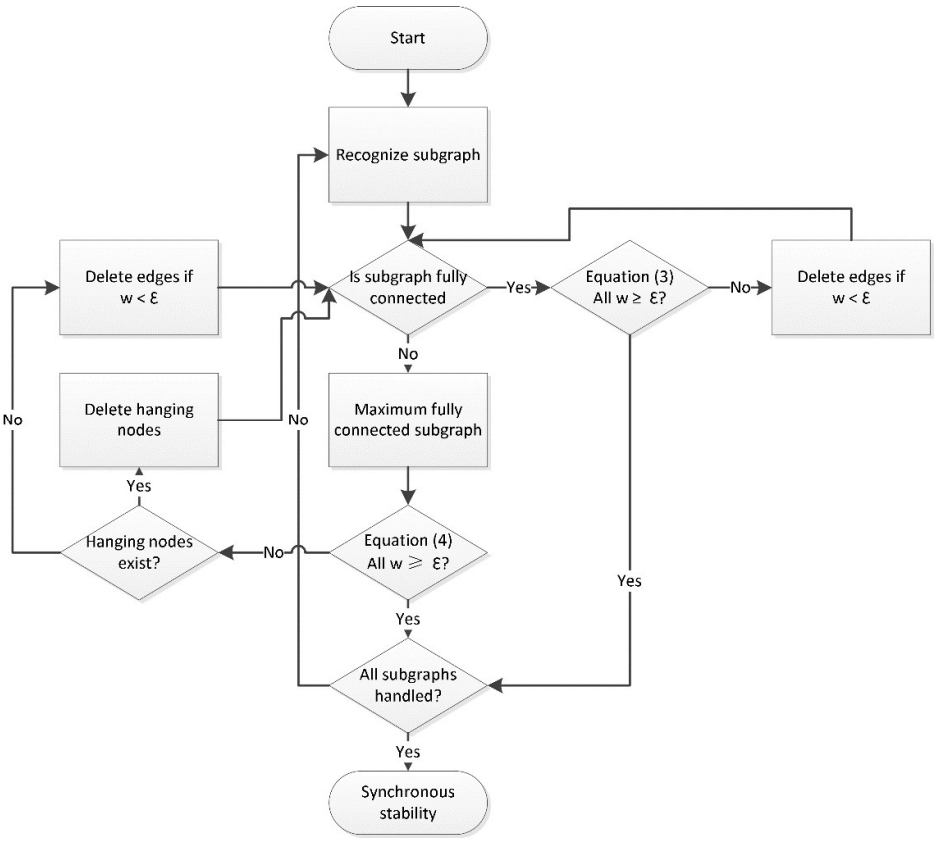


Figure 8. Flowchart followed for synchronously stable network identification.

In the identification process for non-fully connected graphs, the maximum fully connected subgraph had to be obtained. This is called the maximum clique problem and is *NP*-hard that no known algorithms can achieve optimized solution. To solve this problem, some widely used algorithms include the greedy search algorithm, intelligent search algorithm, and heuristic search algorithm. In this work, we chose the greedy search algorithm.

4.2. Evaluation Susceptible Fusion Gene

Here, we describe the algorithm for prioritizing the susceptible cancer fusion gene using graph theory and gene network. First, we estimated the importance of gene nodes in the gene network. Then, the cancer susceptibility of the fusion genes was evaluated based on the importance of the partner gene nodes. The algorithm estimates the importance of gene nodes by evaluating the destructiveness

to the network of deleting the node, where the destructiveness is evaluated according to the difference between the synchronously stable networks before and after the removal of the node. To ensure the gene network stayed synchronously stable, an identification method for synchronously stable networks was used. Figure 8 shows the process followed for achieving a synchronously stable network.

4.2.1. Network Difference Evaluation

The impacts of deleting a node and its associated edges include two aspects: the impact on the degree of remained nodes, and the influence on connectivity. Considering these two aspects, we defined the metric for a gene network, $M(G)$, as the ratio of total edge weights of network to the number of subgraphs:

$$M(G) = \frac{\sum_{i \in G} \sum_{j \neq i \wedge j \in G} w_{ij}}{m_G} \tag{5}$$

where G indicates the gene network, m_G denotes the number of subgraphs of G, and w_{ij} denotes the weight between nodes i and j, $i, j \in G$. Once a node is deleted, the total edge weights decrease and the network connectivity decreases as well, which can be reflected by the increasing of number of subgraphs. All these influences can decrease the value of $M(G)$. The network difference $D(G,v)$ in deleting node v can be represented as:

$$\begin{aligned} D(G,v) &= M(G) - M(G-v) \\ &= \frac{\sum_{i \in G} \sum_{j \neq i \wedge j \in G} w_{ij}}{m_G} - \frac{\sum_{i \in (G-v)} \sum_{j \neq i \wedge j \in (G-v)} w_{ij}}{m_{G-v}} \end{aligned} \tag{6}$$

where $G - v$ is the network obtained by removing node v and its corresponding edges from network G. Based on the difference $D(G,v)$, the importance $H(G,v)$ of node v in network G is defined as:

$$H(G,v) = \frac{D(v)}{M(G_s)} = \frac{M(G_s) - M((G_s - v)_s)}{M(G_s)} \tag{7}$$

where G_s and $(G_s - v)_s$ represent the network G in synchronously stable state depending on whether or not node v is deleted.

4.2.2. Calculation of Gene Node Importance

The algorithm for evaluating a gene node uses the synchronous stability method. Let $G = (V, W)$ represent the gene network. n is the number of nodes in the network, $V = \{v_1, v_2, \ldots, v_n\}$ indicates the set of nodes, and $W = \{w_{01}, w_{02}, \ldots, w_{ij}, \ldots, w_{nk}\}$ represents the set of edge weights, where w_{ij} is the edge weight between nodes i and j, $i,\ j \in V$. The algorithm processes are as follows:

Step 1: Utilize the susceptible cancer gene test data to generate gene network G from the human gene network.

Step 2: Process G by the synchronously stable network identification procedures described in Section 4.1.2., marked as G_s.

Step 3: Delete a node v_i and its associated edges in G_s, $(G_s - \mathrm{v}_i)$.

Step 4: Use the synchronously stable network identification to process $(G_s - \mathrm{v}_i)$ to obtain the synchronously stable state, $(G_s - \mathrm{v}_i)_s$.

Step 5: Use Equation (6) to calculate the $D(G,v)$ value, and subsequently calculate the $H(G,v)$ value using Equation (7).

Step 6: Evaluate the importance of every node in the gene network by repeating Steps 3–5.

4.2.3. Evaluation of Susceptible Cancer Fusion Genes

By using the importance of the partner gene nodes, the significance of the fusion gene could be evaluated. The significance of a fusion gene is calculated by adding partner genes' importance together then multiplying the weight of the edge between two partners. The significance $S(f)$ of fusion gene f is:

$$S(f) = \left(1 + w_{ij}\right)\left(H(i) + H(j)\right) \tag{8}$$

where i and j denote the partner gene node associated with f, w_{ij} is the edge weight between i and j, and $H(i)$ and $H(j)$ are the significance values of i and j, respectively. Fusion genes are formed by the interaction of partner genes. Edge weight between partner gene nodes reflects the interactive relationship between partners genes. Therefore, we consider this probabilistic value when evaluating the fusion gene's significance.

5. Conclusions

This study proposed a method called Synstable Fusion for prioritizing the importance of fusion nodes in a weighted graph, based on the synchronous stability of gene network. This method, when applied to a gene network, effectively evaluates important fusion genes and identifies possible cancer pathogenicity fusion genes. The experimental results showed that the effectiveness of the proposed algorithm is superior to the other two algorithms based on network fusion centrality. In the experiment, we also found some issues that need attention, which could be the focus of future research and development. First, a more accurate gene network generation method should be explored to increase the reliability of the evaluation calculations. Second, other relevant theories can be applied instead of the synchronous stability method to achieve a more efficient and accurate interference information filtering method. In addition, we will try to introduce other algorithms that consider the node's effect on network topology, so that we can more accurately evaluate the value of a node in the network.

Supplementary Materials: The following are available online, Table S1: Lists of genes used in experiment, Table S2: Experimental results of datasets including known oncogenic fusion genes.

Author Contributions: J.W. and X.Z. conducted this study; J.W., A.G., and X.Z. conceived and designed the algorithms; A.G., M.X., and S.W. designed and performed the experiments; M.X., X.Z., Z.Z., and J.W. wrote the manuscript. All authors read and approved the final version of this manuscript.

Funding: This work is supported by the National Science Foundation of China (Grant No: 31701150) and the Fundamental Research Funds for the Central Universities (CXTD2017003). The research funds cover the costs to publish in open access.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mertens, F.; Johansson, B.; Fioretos, T.; Mitelman, F. The emerging complexity of gene fusions in cancer. *Nat. Rev. Cancer* **2015**, *15*, 371–381. [CrossRef] [PubMed]
2. Kumar-Sinha, C.; Kalyana-Sundaram, S.; Chinnaiyan, A.M. Landscape of gene fusions in epithelial cancers: Seq and ye shall find. *Genome Med.* **2015**, *7*, 129. [CrossRef] [PubMed]
3. Latysheva, N.S.; Babu, M.M. Discovering and understanding oncogenic gene fusions through data intensive computational approaches. *Nucleic Acids Res.* **2016**, *44*, 4487–4503. [CrossRef] [PubMed]
4. Persson, H.; Søkilde, R.; Häkkinen, J.; Pirona, A.C.; Vallon-Christersson, J.; Kvist, A.; Mertens, F.; Borg, Å.; Mitelman, F.; Höglund, M.; et al. Frequent miRNA-convergent fusion gene events in breast cancer. *Nat. Commun.* **2017**, *8*, 788. [CrossRef] [PubMed]
5. Lu, C.; Xie, M.; Wendl, M.C.; Wang, J.; McLellan, M.D.; Leiserson, M.D.; Huang, K.L.; Wyczalkowski, M.A.; Jayasinghe, R.; Banerjee, T.; et al. Patterns and functional implications of rare germline variants across 12 cancer types. *Nat. Commun.* **2015**, *6*, 10086. [CrossRef] [PubMed]

6. Huang, K.L.; Mashl, R.J.; Wu, Y.; Ritter, D.I.; Wang, J.; Oh, C.; Paczkowska, M.; Reynolds, S.; Wyczalkowski, M.A.; Oak, N.; et al. Pathogenic Germline Variants in 10,389 Adult Cancers. *Cell* **2018**, *173*, 355–370.e14. [CrossRef] [PubMed]
7. Kim, D.; Salzberg, S.L. TopHat-Fusion: An algorithm for discovery of novel fusion transcripts. *Genome Boil.* **2011**, *12*, R72. [CrossRef] [PubMed]
8. McPherson, A.; Hormozdiari, F.; Zayed, A. deFuse: An Algorithm for Gene Fusion Discovery in Tumor RNA-Seq Data. *PLoS Comput. Boil.* **2011**, *7*, e1001138. [CrossRef] [PubMed]
9. Zhang, J.; White, N.M.; Schmidt, H.K.; Fulton, R.S.; Tomlinson, C.; Warren, W.C.; Wilson, R.K.; Maher, C.A. INTEGRATE: Gene fusion discovery using whole genome and transcriptome data. *Genome Res.* **2016**, *26*, 108–118. [CrossRef] [PubMed]
10. Haber, D.A.; Settleman, J. Cancer: Drivers and passengers. *Nature* **2007**, *446*, 145–146. [CrossRef] [PubMed]
11. Grigoryev, Y.A.; Kurian, S.M.; Avnur, Z.; Borie, D.; Deng, J.; Campbell, D.; Sung, J.; Nikolcheva, T.; Quinn, A.; Schulman, H.; et al. Deconvoluting post-transplant immunity: Cell subset-specific mapping reveals pathways for activation and expansion of memory T, monocytes and B cells. *PLoS ONE* **2010**, *5*, e13358. [CrossRef] [PubMed]
12. Johnson, J.M.; Castle, J.; Garrett-Engele, P.; Kan, Z.; Loerch, P.M.; Armour, C.D.; Santos, R.; Schadt, E.E.; Stoughton, R.; Shoemaker, D.D. Genome-wide survey of human alternative pre-mRNA splicing with exon junction microarrays. *Science* **2003**, *302*, 2141–2144. [CrossRef] [PubMed]
13. Schadt, E.E.; Edwards, S.W.; GuhaThakurta, D.; Holder, D.; Ying, L.; Svetnik, V.; Leonardson, A.; Hart, K.W.; Russell, A.; Li, G.; et al. A comprehensive transcript index of the human genome generated using microarrays and computational approaches. *Genome Biol.* **2004**, *5*, R73. [CrossRef] [PubMed]
14. Wang, J.; Zhao, Z.; Cao, Z.; Yang, A.; Zhang, J. A probabilistic method for identifying rare variants underlying complex traits. *BMC Genomics* **2013**, *14*, S11. [CrossRef] [PubMed]
15. Blomen, V.A.; Májek, P.; Jae, L.T.; Bigenzahn, J.W.; Nieuwenhuis, J.; Staring, J.; Sacco, R.; Diemen, F.R.; Olk, N.; Stukalov, A.; et al. Gene essentiality and synthetic lethality in haploid human cells. *Science* **2015**, *350*, 1092–1096. [CrossRef] [PubMed]
16. Papin, J.A.; Hunter, T.; Palsson, B.O.; Subramaniam, S. Reconstruction of cellular signalling networks and analysis of their properties. *Nat. Rev. Mol. Cell Biol.* **2005**, 99–111. [CrossRef] [PubMed]
17. Wu, G.; Feng, X.; Stein, L. A human functional protein interaction network and its application to cancer data analysis. *Genome Biol.* **2010**, *11*, R53. [CrossRef] [PubMed]
18. Zhou, L.; Lyons-Rimmer, J.; Ammoun, S.; Müller, J.; Lasonder, E.; Sharma, V.; Ercolano, E.; Hilton, D.; Taiwo, I.; Barczyk, M.; et al. The scaffold protein KSR1, a novel therapeutic target for the treatment of Merlin-deficient tumors. *Oncogene* **2016**, *35*, 3443–3453. [CrossRef] [PubMed]
19. Soler-López, M.; Zanzoni, A.; Lluís, R.; Stelzl, U.; Aloy, P. Interactome mapping suggests new mechanistic details underlying Alzheimer's disease. *Genome Res.* **2011**, *21*, 364–376. [CrossRef] [PubMed]
20. Rodgers-Melnick, E.; Culp, M.; DiFazio, S.P. Predicting whole genome protein interaction networks from primary sequence data in model and non-model organisms using ENTS. *BMC Genom.* **2013**, *14*, 608. [CrossRef] [PubMed]
21. Stark, C.; Breitkreutz, B.J.; Reguly, T.; Boucher, L.; Breitkreutz, A.; Tyers, M. BioGRID: A general repository for interaction datasets. *Nucleic Acids Res.* **2006**, *34*, D535–D539. [CrossRef] [PubMed]
22. Zeng, X.; Liu, L.; Lü, L.; Zou, Q. Prediction of potential disease-associated microRNAs using structural perturbation method. *Bioinformatics* **2018**, *34*, 2425–2432. [CrossRef] [PubMed]
23. Geng, Y.; Zhao, Z.; Zhang, X.; Wang, W.; Cui, X.; Ye, K.; Xiao, X.; Wang, J. An improved burden-test pipeline for identifying associations from rare germline and somatic variants. *BMC Genomics* **2017**, *18*, 55–62. [CrossRef] [PubMed]
24. Wang, H.; Ding, S.; Wu, D.; Zhang, Y.; Yang, S. Smart connected electronic gastroscope system for gastric cancer screening using multi-column convolutional neural networks. *Int. J. Prod. Res.*. [CrossRef]
25. Warde-Farley, D.; Donaldson, S.L.; Comes, O.; Zuberi, K.; Badrawi, R.; Chao, P.; Franz, M.; Grouios, C.; Kazi, F.; Lopes, C.T.; et al. The GeneMANIA prediction server: Biological network integration for gene prioritization and predicting gene function. *Nucleic Acids Res.* **2010**, *38*, W214–W220. [CrossRef] [PubMed]
26. Cantini, L.; Medico, E.; Fortunato, S.; Caselle, M. Detection of gene communities in multi-networks reveals cancer drivers. *Sci. Rep.* **2015**, *5*, 17386. [CrossRef] [PubMed]

27. Cava, C.; Bertoli, G.; Colaprico, A.; Olsen, C.; Bontempi, G.; Castiglioni, I. Integration of multiple networks and pathways identifies cancer driver genes in pan-cancer analysis. *BMC Genom.* **2018**, *19*. [CrossRef] [PubMed]
28. Freeman, L.C. Centrality in Social Networks Conceptual Clarification. *Soc. Netw.* **1978**, *1*, 215–239. [CrossRef]
29. Freeman, L.C. A Set of Measures of Centrality Based on Betweenness. *Sociometry* **1977**, *40*, 35–41. [CrossRef]
30. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Netw.* **2015**, *61*, 85–117. [CrossRef] [PubMed]
31. Palanisamy, N.; Ateeq, B.; Kalyana-Sundaram, S.; Pflueger, D.; Ramnarayanan, K.; Shankar, S.; Han, B.; Cao, Q.; Cao, X.; Suleman, K.; et al. Rearrangements of the RAF kinase pathway in prostate cancer, gastric cancer and melanoma. *Nat. Med.* **2010**, *16*, 793–798. [CrossRef] [PubMed]
32. Robinson, D.R.; Kalyana-Sundaram, S.; Wu, Y.M.; Shankar, S.; Cao, X.; Ateeq, B.; Asangani, I.A.; Iyer, M.; Maher, C.A.; Grasso, C.S.; et al. Functionally recurrent rearrangements of the MAST kinase and Notch gene families in breast cancer. *Nat. Med.* **2011**, *17*, 1646–1651. [CrossRef] [PubMed]
33. Wang, X.S.; Prensner, J.R.; Chen, G.; Cao, Q.; Han, B.; Dhanasekaran, S.M.; Ponnala, R.; Cao, X.; Varambally, S.; Thomas, D.G.; et al. An integrative approach to reveal driver gene fusions from paired-end sequencing data in cancer. *Nat. Biotechnol.* **2009**, *27*, 1005–1011. [CrossRef] [PubMed]
34. Wu, C.C.; Kannan, K.; Lin, S.; Yen, L.; Milosavljevic, A. Identification of cancer fusion drivers using network fusion centrality. *Bioinformatics* **2013**, *29*, 1174–1181. [CrossRef] [PubMed]
35. Belykh, V.N.; Belykh, I.V.; Hasler, M. Connection graph stability method for synchronized coupled chaotic systems. *Phys. D Nonlinear Phenom.* **2004**, *195*, 159–187. [CrossRef]
36. Wu, C.C.; Asgharzadeh, S.; Triche, T.J.; D'Argenio, D.Z. Prediction of human functional genetic networks from heterogeneous data using RVM-based ensemble learning. *Bioinformatics* **2010**, *26*, 807–813. [CrossRef] [PubMed]
37. He, L.; Wang, Y.; Yang, Y.; Huang, L.; Wen, Z. Identifying the gene signatures from gene-pathway bipartite network guarantees the robust model performance on predicting the cancer prognosis. *Biomed. Res. Int.* **2014**, *2014*, 424509. [CrossRef] [PubMed]
38. Wang, H.; Huang, L.; Jing, R.; Yang, Y.; Liu, K.; Li, M.; Wen, Z. Identifying oncogenes as features for clinical cancer prognosis by Bayesian nonparametric variable selection algorithm. *Chemom. Intell. Lab. Syst.* **2015**, *146*, 464–471. [CrossRef]
39. Grover, M.P.; Ballouz, S.; Mohanasundaram, K.A.; George, R.A.; Sherman, C.D.; Crowley, T.M.; Wouters, M.A. Identification of novel theracassociation data. *BMC Med. Genom.* **2014**, *7* (Suppl. S1), S8. [CrossRef] [PubMed]
40. Schneider, L.; Stöckel, D.; Kehl, T.; Gerasch, A.; Ludwig, N.; Leidinger, P.; Huwer, H.; Tenzer, S.; Kohlbacher, O.; Hildebrandt, A.; et al. DrugTargetInspector: An assistance tool for patient treatment stratification. *Int. J. Cancer* **2016**, *138*, 1765–1776. [CrossRef] [PubMed]
41. Makhijani, R.K.; Raut, S.A.; Purohit, H.J. Identification of common key genes in breast, lung and prostate cancer and exploration of their heterogeneous expression. *Oncol. Lett.* **2018**, *15*, 1680–1690. [CrossRef] [PubMed]
42. Abate, F.; Zairis, S.; Ficarra, E.; Acquaviva, A.; Wiggins, C.H.; Frattini, V.; Lasorella, A.; Iavarone, A.; Inghirami, G.; Rabadan, R. Pegasus: A comprehensive annotation and prediction tool for detection of driver gene fusions in cancer. *BMC Syst. Boil.* **2014**, *8*, 97. [CrossRef] [PubMed]
43. Zhao, J.; Li, X.; Yao, Q.; Li, M.; Zhang, J.; Ai, B.; Liu, W.; Wang, Q.; Feng, C.; Liu, Y.; et al. RWCFusion: Identifying phenotype-specific cancer driver gene fusions based on fusion pair random walk scoring method. *Oncotarget* **2016**, *7*, 61054–61068. [CrossRef] [PubMed]
44. Gu, J.; Chukhman, M.; Lu, Y.; Liu, C.; Liu, S.; Lu, H. RNA-seq Based Transcription Characterization of Fusion Breakpoints as a Potential Estimator for Its Oncogenic Potential. *BioMed Res. Int.* **2017**, *2017*, 9829175. [CrossRef] [PubMed]
45. Frenkel-Morgenstern, M.; Gorohovski, A.; Tagore, S.; Sekar, V.; Vazquez, M.; Valencia, A. ChiPPI: A novel method for mapping chimeric protein–protein interactions uncovers selection principles of protein fusion events in cancer. *Nucleic Acids Res.* **2017**, *45*, 7094–7105. [CrossRef] [PubMed]
46. Hu, X.; Wang, Q.; Tang, M.; Barthel, F.; Amin, S.; Yoshihara, K.; Lang, F.M.; Martinez-Ledesma, E.; Lee, S.H.; Zheng, S.; et al. TumorFusions: An integrative resource for cancer-associated transcript fusions. *Nucleic Acids Res.* **2018**, *46*, D1144–D1149. [CrossRef] [PubMed]

47. Futreal, P.A.; Coin, L.; Marshall, M.; Down, T.; Hubbard, T.; Wooster, R.; Rahman, N.; Stratton, M.R. A census of human cancer genes. *Nat. Rev. Cancer* **2004**, *4*, 177–183. [CrossRef] [PubMed]
48. Tipping, M.E.; Smola, A. Sparse Bayesian Learning and the Relevance Vector Machine. *J. Mach. Learn. Res.* **2001**, *1*, 211–244. [CrossRef]
49. Tsechansky, M.S.; Provost, F. Handling Missing Values when Applying Classification Models. *J. Mach. Learn. Res.* **2007**, *8*, 1625–1657.
50. Liu, H.; Cao, M.; Wu, C.W. Graph comparison and its application in network synchronization. In Proceedings of the 12th European Control Conference, Zurich, Switzerland, 17–19 July 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 3809–3814.

Sample Availability: Not available.