基于图论知识模拟

蛋白质间的相互作用关系

**一、背景及问题引入**

蛋白质是由一个或多个氨基酸残基长链组成的生物大分子，其作为生命体的重要组成物质，承担着一系列重要的生命活动。蛋白质大分子通常通过与其它蛋白质的作用来行使功能。蛋白质在生物体内扮演着多种功能角色，包括信号传导、代谢调控、细胞结构维持等。了解蛋白质之间的相互作用关系有助于揭示生物体内复杂的生物过程、细胞信号。

在生物学研究中，科学家经常关注蛋白质之间的相互作用关系，这对于理解细胞内的生物过程、疾病机制等具有重要意义。蛋白质相互作用网络是一种用于描述和分析这些关系的工具。通过构建蛋白质相互作用网络，可以揭示哪些蛋白质相互作用，从而形成一个更全面的生物学图景。

所以，这篇报告将会讨论如何用图来模拟蛋白质之间的相互作用关系，以及用图的算法来表示图的性质的指标，以方便使用者的研究。

1. **图的建模**
2. **蛋白质信息的收集**

信息收集涉及数据库的操作，由于知识体系庞大，现使用STRING数据库来实现蛋白质信息的收集。

STRING数据库介绍： STRING数据库是一个基于公共数据库和文献信息的蛋白质相互作用网络数据库。它收集了多个公共数据库，包括UniProt、KEGG、NCBI和Gene Ontology等，整合了这些数据并生成一个全面的蛋白质相互作用网络数据库。STRING数据库收集了大量已发表的实验结果，实验结果来自于多种实验技术，包括质谱法、酵母双杂交法、蛋白质微阵列等。

数据库访问：用户可以通过STRING数据库的官方网站（[https://string-db.org/](https://link.zhihu.com/?target=https://string-db.org/" \t "https://zhuanlan.zhihu.com/p/_blank)）访问该数据库，并使用搜索工具输入感兴趣的蛋白质或者基因的名字，将会得到与查询结果相关的蛋白质相互作用网络和其他相关信息。

1. **图的节点与边的定义**

在蛋白质相互作用网络中，图的节点和边的定义是关键的，这决定了网络的拓扑结构和表示。

**① 节点：**

节点表示蛋白质：每个蛋白质被表示为图中的一个节点。这个节点可以使用蛋白质的名称来标识。每个节点通常会附带一些属性信息，例如蛋白质的功能、亚细胞定位、蛋白质的空间结构等。

**② 边：**

边表示相互作用：如果两个蛋白质之间存在相互作用，就在它们之间添加一条边。这样，整个网络就是一个图，其中边表示蛋白质之间的相互作用。在设计UI界面时，可以改变线条颜色，通过不同颜色来表示不同类型的相互作用。

而边上可能有权重，表示相互作用的强度或可靠性。这些权重基于生理学上对相互作用的划分，优先进行相互作用，或者相互作用强烈的，可以赋给更高的权值。

边还应具有方向性，在一些网络中，边是有向的，表示相互作用的方向。例如，蛋白质A影响蛋白质B，但不一定反之。所以，可以采用有向图来实现。

**③ 图的存储：**

图有两种常用存储结构，一种是邻接矩阵，另一种是邻接表。邻接矩阵有一些缺点，当顶点很多且边比较少时，比较浪费空间，并且两个节点之间的路径不好求，若要确定图中有多少条边，需要遍历一遍邻接矩阵。故选择邻接表来实现图的存储，在节点中写入节点的属性和边的权值。

1. **子图**

当我们储存了蛋白质的相互作用信息，我们要选择我们想要查看的蛋白质，也就是在图中找到对应的子图，这个我们采用点导出子图的方法。

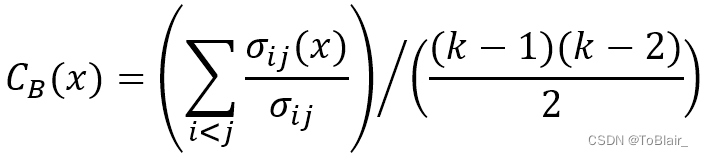
方法：设S是V(G)的[子集](https://so.csdn.net/so/search?q=%E5%AD%90%E9%9B%86&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/weixin_44925715/article/details/_blank)，以S为点集，以G的所有那些两端点都在S内的边组成边集，所得到的G的子图称为S在G中的导出子图，或更确切地，节点导出子图。（一个顶点子集以及两个端点都在这个子集中的所有边构成的子图）

1. **图的中心性分析**

在蛋白质相互作用图中，中心性指标用于研究网络中蛋白质节点的相对重要性和在生物学系统中的功能。不同的中心性指标提供了关于蛋白质网络结构和功能的不同信息，有助于识别网络中的关键节点。

①中介中心性用来衡量点处于其它任意两点间最短路径之中的概率。该概念由 Linton C. Freeman 于 1977 年提出，能有效地计算出在图的多个部分间起重要桥梁、媒介作用的点。所以这个方法很适用于找到比较重要的蛋白质，方便研究蛋白之间的关系与作用。

中介中心性的取值范围是 [0,1]，数值越大中介作用越强。



②我们也可以不这么复杂，直接使用节点度来近似表示蛋白质的相互作用数量即可，在查询结果中按顺序输出节点度的大小和蛋白质名称即可。

1. **直径和平均最短路径长度的应用**

这一部分就是使用直径和平均最短路径长度来分析图的性质，以便于研究。

图的直径是图中最长路径的长度，即网络中任意两个节点之间最远的距离。用处：直径提供了网络中最大的信息传递距离。在蛋白质相互作用图中，直径可以帮助了解蛋白质之间可能的远距离调控关系。较大的直径可能表示网络中存在长程的信号传导路径，也可能提示网络的整体规模和复杂性。

平均最短路径长度是图中所有节点对之间最短路径长度的平均值。用处：平均最短路径长度反映了网络中信息传递的整体效率。在蛋白质相互作用图中，较小的平均最短路径长度可能表明蛋白质之间的相互作用关系相对紧密，信息传递更加迅速和高效。

这两个方法适用于反映蛋白质相互作用关系的整体情况，可以帮助使用者更深入地理解蛋白质相互作用网络的结构和功能，为研究生物学过程提供关键信息。

1. **算法分析**

1.数据获取和预处理：

从生物信息学数据库中获取蛋白质相互作用数据。针对数据格式和结构进行预处理，将数据转换为图的表示形式。

2.图的构建：

调用图论库构建蛋白质相互作用图。将蛋白质表示为图的节点，相互作用关系表示为图的边，边有向，并赋有权值。邻接表来实现图的存储。

3.图的分析：

根据输入来提取出子图，计算图的直径、平均最短路径长度，以了解网络的整体特征。使用中心性算法（度中心性、介数中心性）识别网络中的重要节点。下面是一个计算样例函数：

private static void calculateGraphMetrics(Graph<String, DefaultEdge> graph) {

GraphMeasurer<String, DefaultEdge> graphMeasurer = new GraphMeasurer<>(graph);

// 计算图的直径

double diameter = graphMeasurer.getDiameter();

System.out.println("\n图的直径: " + diameter);

// 计算平均最短路径长度

double avgShortestPathLength = graphMeasurer.getGraphDiameter();

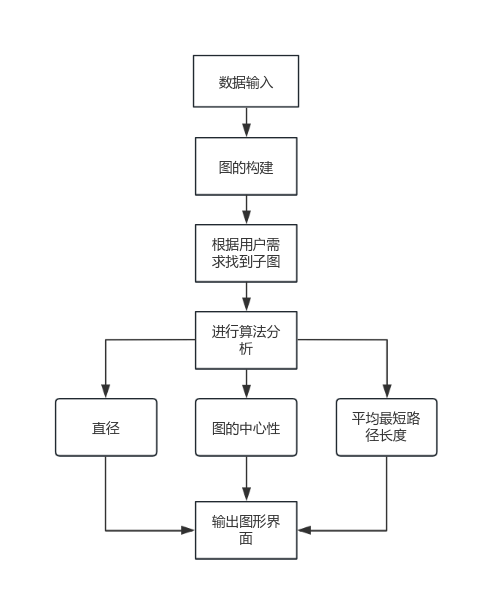
System.out.println("图的平均最短路径长度: " + avgShortestPathLength);

}

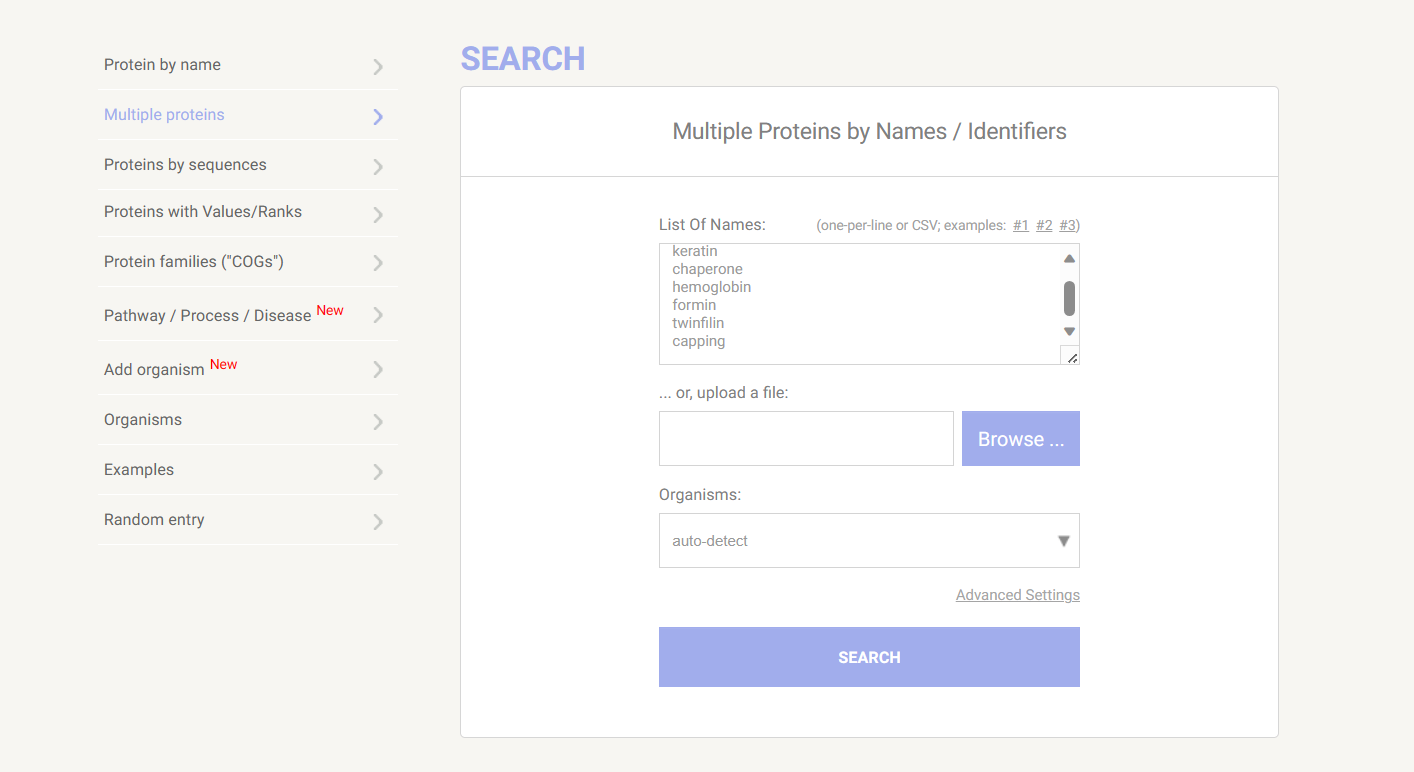
}

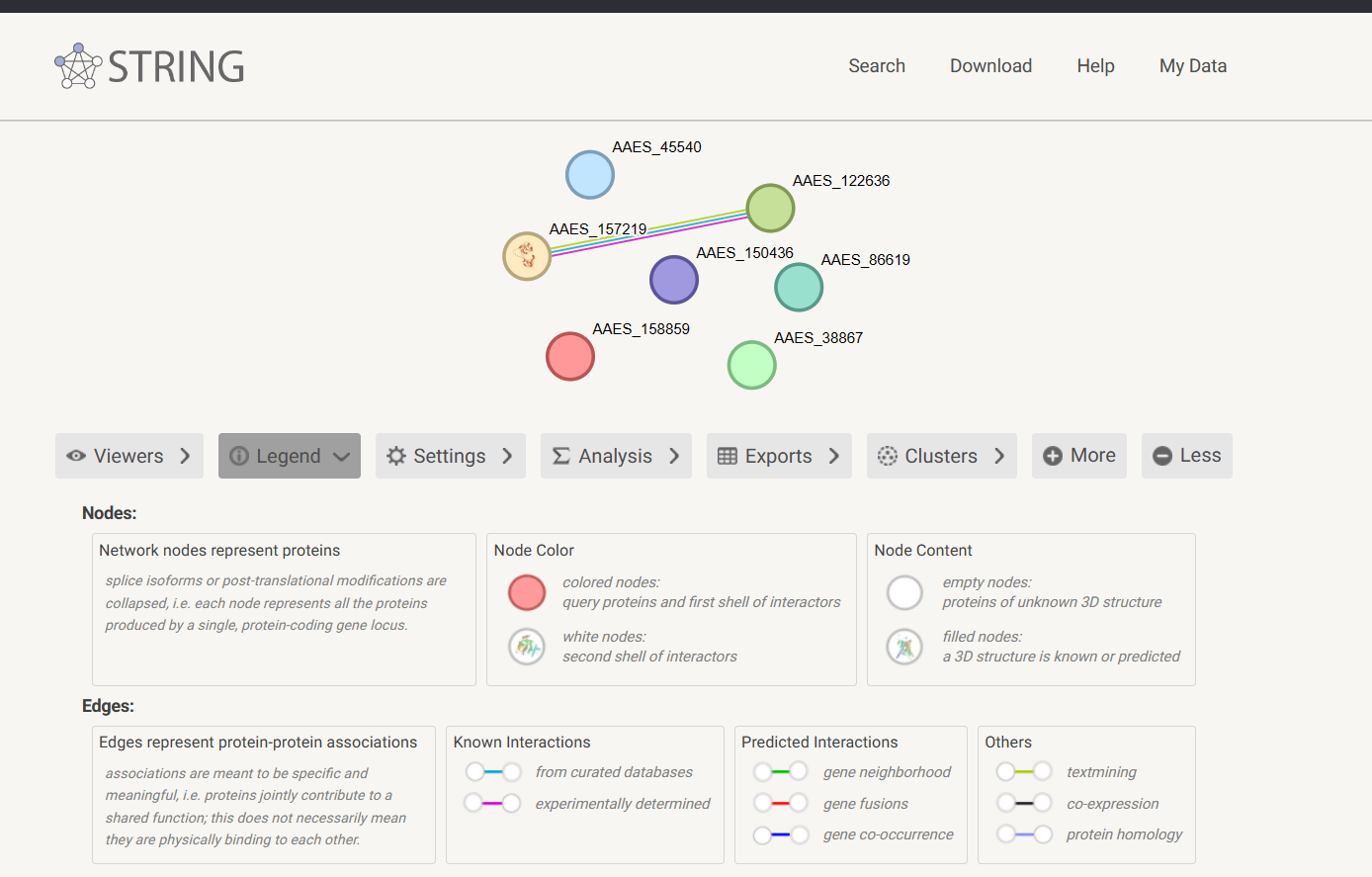
4.图的表示

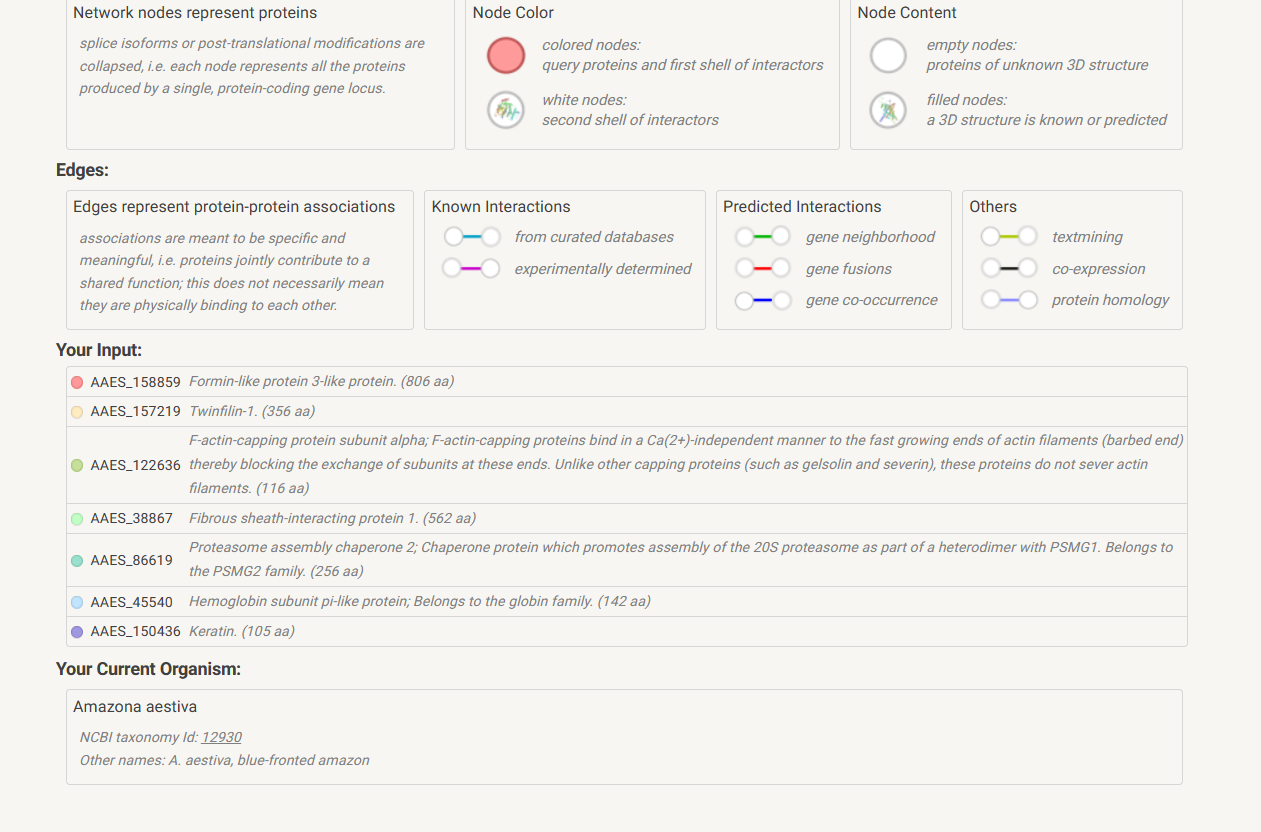
使用可视化工具将图形表示出来，以直观地展示蛋白质相互作用网络的结构和特性。在Java中，可以使用 JGraphX 库来进行蛋白质相互作用图的可视化。创建蛋白质相互作用图，创建JGraphX适配器，使用mxCircleLayout布局，创建图形组件，创建并显示JFrame窗口等操作，可以做出简单的图形界面。

****

1. **STRING数据库成果展示**







**参考文献：**

1. 《STRING数据库：蛋白质相互作用(PPI网络)的分析神器~ 》——云生信学生物信息

<https://zhuanlan.zhihu.com/p/627355726> 来源：知乎

2. 《基于蛋白质相互作用网络的分析算法的研究》

[扬州大学](https://kns.cnki.net/kcms2/organ/detail?v=p7sfyaWOx3NT29EnwpipZz3R0CA2EAVAxUZd9QQm1ERe7b1GIU0ODU4tU1EtN5GdvudY9PjD2QIT3fFRJzfcFQMOe1gCbguQ0OOTGyDJg50=&uniplatform=NZKPT" \t "https://kns.cnki.net/kcms2/article/_blank) [吴蔷梅](https://kns.cnki.net/kcms2/author/detail?v=p7sfyaWOx3MXt9x35-QBofoKbyIwzEvmqeqY6G9YxfJ4554y_2UAuHOSgZCsqCO5rNFw0RcCzrSzpNlA1nEZTTgnHSqMraYK3whrtInFtZk=&uniplatform=NZKPT" \t "https://kns.cnki.net/kcms2/article/_blank) 来源：中国知网

1. 《【图论】节点的几种中心性》 —— ToBlair\_

<http://t.csdnimg.cn/htymU> 来源：CSDN

1. STRING数据库 <https://string-db.org/>
2. 《离散数学（第三版）》 西安电子科技大学出版社
3. 《数据结构（C语言版）》 清华大学出版社