**Санкт-Петербургский государственный университет**
**математико-механический факультет**

Кафедра системного программирования

# Выделение научных сообществ на основе анализа библиографических данных

Курсовая работа студента 345 группы
Филатова Владимира Константиновича

Научный руководитель, ст. преп. …… Владимир Суворов
EMC Санкт-Петербург

Санкт-Петербург

2012

# Оглавление

# Введение

Современное общество постоянно использует интернет не только для того, чтобы получать некоторую информацию, но и для того, чтобы делиться своей. На данный момент есть много социальных сетей, видео хостингов, файловых хостингов и других различных ресурсов сети с большими объемами информации и базами данных. Количество таких ресурсов растет постоянно, так как меняется мир и меняются информационные технологии. Не понятно, что нас ждет через 10, 15 или 20 лет. Люди каждый год придумывают что-то новое. Где-то экономят на памяти, где-то на пространстве хранения и т.п. Но самое главное, большинство людей оставляют подробную информацию о себе на многих ресурсах. Эта информация может быть об увлечениях человека, его образовании, его местах путешествий и других сферах его деятельности Поэтому многие люди и компании решили выделять информацию из таких ресурсов.

В связи с большим количеством таких компаний и ресурсов, появилось новое направление в компьютерной индустрии, такое как data mining. Этот термин ввели еще в 1989 году, и он предполагает выделение некоторых «скрытых» данных из уже имеющихся.

Основу методов Data Mining составляют всевозможные методы классификации, моделирования и прогнозирования, основанные на применении деревьев решений, искусственных нейронных сетей, генетических алгоритмов, эволюционного программирования, ассоциативной памяти, нечёткой логики. К методам Data Mining нередко относят *статистические методы* (дескриптивный анализ, корреляционный и регрессионный анализ, факторный анализ, дисперсионный анализ, компонентный анализ, дискриминантный анализ, анализ временных рядов). Такие методы, однако, предполагают некоторые априорные представления об анализируемых данных, что несколько расходится с целями *Data Mining* (обнаружение ранее неизвестных нетривиальных и практически полезных знаний).

Подразделом этого направления является направление community mining, которое предполагает выделение сообществ людей из базы информации о них. Например, из какой-нибудь социальной сети можно выделить людей работающих в одной компании, или людей болеющих за одну футбольную команду и т.п. Причем выделять сообщества можно не только из социальных сетей.

Большим развивающимся компаниям необходимо знать кучу информации о своих сотрудниках, так как все работники должны работать в комфортных для них условиях, чтобы эффективность труда была высокая. Например, если в компании все люди увлекаются футболом, то можно организовать турнир между отделами, или сделать сборную команду для игр против других компаний. Если часть сотрудников любит выезжать на природу, то можно сделать корпоративные выезды.

Направление community mining занимается сбором такой статистики. Все сотрудники могут оставлять ссылки на свои странички в социальных сетях, и компания может искать необходимые группы людей по какому-нибудь направлению.

# Постановка задачи

Научное сообщество - группа людей, которая участвует в разработке определённой предметной области или проблемы. В него могут входить соавторы статей, а также авторы статей, на которые ссылается какая-нибудь статья по данной теме. На данный момент таких сообществ большое количество. Поиск таких сообществ нужен тем людям, которые собираются начать писать новую научную статью или начать новое исследование в каком-нибудь научном направлении.

Если между двумя авторами окажется очень много связей, это значит, что они практически всегда работают над какой-нибудь темой вместе.

Благодаря выделению сообществ можно найти человека или группу людей, для того чтобы начать исследование в соавторстве с кем-нибудь или узнать что было сделано, что можно сделать или придумать что-нибудь из того, что можно было бы сделать по данной теме или направлении.

Моей задачей является создание удобного приложения (web сервиса) для выделения научных сообществ, описанных выше. Для основы я буду брать данные, которые располагаются на сайте CiteSeerx (http://citeseer.ist.psu.edu). Он предоставляет информацию об авторах, их публикациях и перекрестных ссылок статей. Результатом будет являться приложение или сервис, который по ключевым словам будет отображать сообщества людей по некоторой проблематике.

# Обзор существующих решений

На данный момент есть несколько ресурсов, которые предлагают услуги поиска статей по названию или поиска автора по именам и фамилиям. Одними из них также являются CiteSeer.

CiteSeerx является развивающейся научной цифровой библиотекой и поисковой машиной, которая прежде всего хранит статьи на темы в области компьютерных и информационных наук. CiteSeerx направлена на улучшение распространения научной литературы и обеспечивает хорошую функциональность, удобство, доступность, полноту, оперативность и своевременность доступных научных и академических знаний.

Вместо того, чтобы создавать новые цифровой библиотеки, CiteSeerx стремится предоставить ресурсы, такие как алгоритмы, данные, метаданные, услуги, методы и

программы, которые могут быть использованы для продвижения других цифровых библиотек. CiteSeerx разработала новые методы и алгоритмы для индексирования PostScript и PDF научные статьи в Интернете.

Возможности CiteSeerX:

1) Автономное индексирование цитирования (ACI) - CiteSeerX использует ACI для автоматического создания индекса цитирования, который может быть использован для поиска литературы и оценки.

2) Цитирования статистика - CiteSeer вычисляет статистику цитирования и связанные документы для всех статей, указанных в базе данных, а не только индексированные статьи.

3) Ссылочная связь - Как и во многих других цифровых библиотеках , CiteSeer позволяет просматривать базу данных, используя ссылки цитирования. Тем не менее, CiteSeer выполняет это автоматически.

4) Цитирование контекстное - CiteSeer может показать контекст ссылок по данной работе, что позволяет исследователю быстро и легко видеть, что другие исследователи вызывают к статье интерес.

5) Связанная документация - CiteSeer находит связанные документы, используя цитаты и слова, и на их основе измеряет и показывает активную и постоянно обновляющуюся библиографию для каждого документа.

6) Мощный поиск - CiteSeer использует направила поиска все более сложные запросы, а также позволяет использование авторских инициалов, чтобы обеспечить более гибкий поиск по имени.

Кроме CiteSeerx в сети есть много других библиотек. Среди них есть библиотека ACM (dl.acm.org), CiNii (http://ci.nii.ac.jp/), и много других, которые используют свои методы поиска статей в интернете, методы индексации цитирования, свои статистики и т.д. Все они имеют довольно удобный интерфейс поиска статей или информации об авторах. Но выделять научные сообщества из них очень сложно.

Чтобы разобраться в связях между авторами, приходится выделять много времени, потому что для того, чтобы выделить какое-нибудь научное сообщество, приходится просматривать кучу статей. Иногда связи могут быть очень запутанными, а на то чтобы распутать их приходится выделять еще больше времени.

В сети уже есть один готовый визуализатор научных сообществ, написанный Ryutaro Ichise и Hideaki Takeda из National Institute of Informatics2-1-2 Hitotsubashi Chiyoda-ku, который бывает доступен по ссылке. Но ссылка периодически бывает недоступна. Эти ученые написали небольшую статью о своем приложении. Они выделяли 3 основных группы связей между авторами: соавторство, цитирование и социтирование. Данные брали из цифровой библиотеки CiNii. Общая схема их приложения указана на рисунке 2. Она не выглядит очень сложной. Состоит из двух баз – одна основная (цифровая библиотека CiNii), а вторая – база данных, с которой общается приложение ( как они её назвали – экспериментальная)

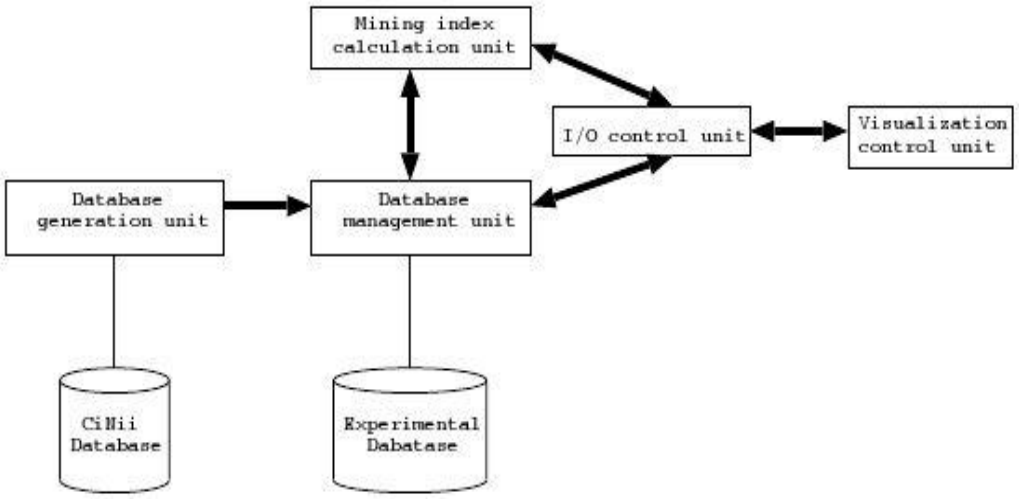


*Рисунок 1.*

В связи постоянной недоступностью этого захотелось сделать нечто похоже на это, а затем его усовершенствовать, добавляя новые методы кластеризации, группировки и другие алгоритмы для наглядного отображения научных сообществ

# Этапы построения приложения

В качестве основного языка программирования был выбран язык Java, так как он является платформенно независимым, объектно-ориентированным языком программирования.

## Построение базы данных.

В связи с тем, что мне пришлось постоянно работать с большими объемами информации, возникла необходимость в построении базы данных. В качестве СУБД было решено взять MYSQL, так как она является наиболее приспособленной для web приложений. Также преимуществами MYSQL являются многопоточность, быстрая работа, масштабируемость, бесплатность, интерфейс с языком Java другими языками.

База данных состоит из 7 таблиц, основными из которых являются:

1) Authors – информация об авторах
2) Papers – информация о статьях
3) Author_Paper – таблица, соединяющая ID статей и ID их авторов
4) Citation – таблица соединяющая ID статей и ID статей, ссылающихся на них

Общую схему бд можно посмотреть на рисунке ниже.

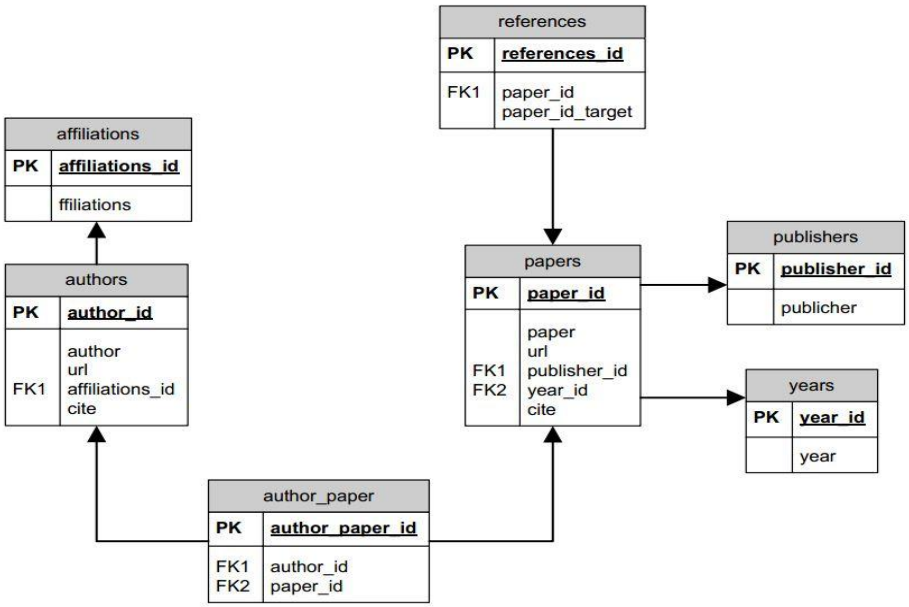


*Рисунок 2.*

## Парсинг сайта CiteSeer.

Для того, чтобы заполнить базу данных, описанную выше, необходимо где-то добыть данные. Было решено взять ресурс CiteSeer и выделить из него всю информацию по статьям и авторам, хранящимся на нем.

В ходе работы были протестированы три библиотеки java парсеров написанных на Java:

1) JSOUP
2) HTML cleaner
3) JERICHO

Jericho - на мой взгляд, худшая из всех этих трех библиотек, так как она работает очень долго и в режиме отладки не показывает дерево разбора, которое получила. Это в свою очередь не дает программисту просмотреть правильность действий, которые он делает, и поэтому поиск ошибки в программе может быть очень долгим, и это при том, что работает она дольше всех.

HTML Cleaner и JSOUP – хорошие библиотеки, так как работают быстро и строят очень хорошее дерево разбора. Обе библиотеки обладают практически одинаковым набором методов. Но HTML Cleaner не смогла распарсить CiteSeerx, в то время как JSOUP сделал это. Однако, например, чтобы разобрать страничку Youtube, для JSOUP пришлось исходный код страницы скопировать в файл, и только затем она смогла распарсить текст,

в то время как HTML Cleaner быстро распарсил ту же самую страничку, не скачивая исходный код в файл.

Поэтому, резюмируя, можно сказать, что выбирать HTMLCleaner или JSOUP нужно в зависимости от сайта, который необходимо разобрать.

Так как HTML Cleaner не справилась с Citeseerx, а JSOUP смогла сделать это, поэтому данные я доставал, используя библиотеку JSOUP.

# Выделение научных сообществ

Когда получили некоторый набор данных, было решено начать выделять научные сообщества. Выделение происходило с помощью Java библиотеки JUNG.

JUNG - это программная библиотека, которая предоставляет расширенный язык для моделирования, анализа и визуализации данных, которые могут быть представлены в виде графа или сети. Она написана на Java, что позволяет использовать обширные встроенные возможности Java API, а также других существующих сторонних библиотек Java.

Архитектура JUNG предназначена для поддержки различных представлений субъектов и их отношений, такие, как направленного и неориентированных графов, мультимодальные графы, графы с параллельными краями и гиперграфов. Она представляет механизм для аннотирования графиков, сущностей, и отношений с метаданными. Это облегчает создание аналитических инструментов для сложных наборов данных, которые можно исследовать отношения между сущностями.

JUNG включает в себя реализацию ряда алгоритмов из теории графов, анализа данных и анализ социальных сетей, таких как программы для кластеризации, декомпозиция, оптимизация, генерация случайных графов, статистический анализ и расчет расстояний сети.

Благодаря библиотеке JUNG по запросу на какую-нибудь тему удалось построить граф научных сообществ. В дальнейшем планируется заняться группировкой и кластеризацией научных сообществ для более детального их отображения.

На сервере формированием дерева занимается сервлет, который при построении графа сообществ передает его апплету, который находится в браузере клиента.

В получившемся дереве вершинами являются авторы, который находятся по запросу из базы данных. Также в графе было два типа ребер, который обозначали соавторство и цитирование в статье одного автора статьи другого автора.

Благодаря такому выделению по некоторым запросам получились интересные картинки, из которых можно было выделить людей принадлежащих к одному научному сообществу.

Алгоритм группировки, используемый в данной работе, основан на варианте алгоритма Брона-Кербоша с дополнением “with pivot vertex”. В алгоритме используется 3 множества вершин R, P и X. Алгоритм находит максимальные клики,

которые содержат вершины из R, некоторые вершины из P и не содержит вершин из X. Дополнение алгоритма “with pivot vertex u from set P” было сделано авторами данного алгоритма для уменьшения количества рекурсивных вызовов алгоритма, так как максимальная клика содержит либо вершину u, либо любую вершину из множества P\N(u). Тем самым отсекались ветви решения с изначально не максимальными кликами. Ниже приведен псевдокод алгоритма с дополнением “with pivot vertex”.

```
BronKerbosch(R,P,X):
 if P and X are both empty:
    report R as a maximal clique
 choose a pivot vertex u in P ⋃ X
 for each vertex v in P \ N(u):
     BronKerbosch(R ⋃ {v}, P ⋂ N(v), X ⋂ N(v))
     P := P \ {v}
     X := X ⋃ {v}
```

На входе алгоритма граф G = < V’, E >, где V’ – это множество вершин-авторов, полученных из базы данных по поисковому запросу пользователя, E – множество ребер, обозначающих связи между этими авторами. Результат работы алгоритма – это граф с выделенными другим цветом максимальными кликами. Дополнительно другим цветом выделены вершины, смежные хотя бы с одной вершиной из клики. В контексте социального графа это авторы, не входящие в научное сообщество, но связанные с ним.

## Отображение научных сообществ.

Благодаря возможностям Java было решено сделать web сервис для отображения научных сообществ, причем делать это через Java апплеты.

Преимущество Java апплетов:

А) кроссплатформенность
Б) поддержка большинство браузеров
В) иметь доступ к машине, если пользователь согласен на это
Г) работа на всех версиях Java
Д) работает с системой сервер/клиент.

Дальнейшая визуализация графа сообществ на апплете также происходит с помощью библиотеки JUNG, которая предоставляет многочисленные возможности по визуализации графов.

Для наглядности, ребра соавторства и цитирования рисуются разным цветом, чтобы легче было понять, в каких масштабах авторы взаимодействуют между собой. Например, если между двумя вершинами куча ребер красного цвета (что означает, что они являются соавторами по одной или нескольким статьям), то можно понять, что авторы постоянно пишут статьи вместе, что значит, что они постоянно общаются.

# Заключение

В ходе данной курсовой работы было написано приложение, которое отображает научные сообщества по запросам на научные темы. Результаты этого приложения можно посмотреть на рисунках 3, 4, 5, 6. Скорость работы данного приложения пока не высокая, так как основной целью было создание основы для web сервиса поиска научных сообществ.

Дальнейшее развитие данного приложения включает:

1) Добавление новых цифровых библиотек в базу данных приложения
2) Ускорение формирования графа научных сообществ
3) Ускорение передачи информации между сервером и апплетом
4) Добавление новых графических отображений графа научных сообществ
5) Добавление методов группировки и кластеризациии.

Примеры работы приложения

**Пример 1.**На рисунке 3, представлен пример работы приложения по поиску научных сообществ, которые можно выделить из экспериментальной базы (не полностью заполненной) научных статей и их авторов по теме «uwb»

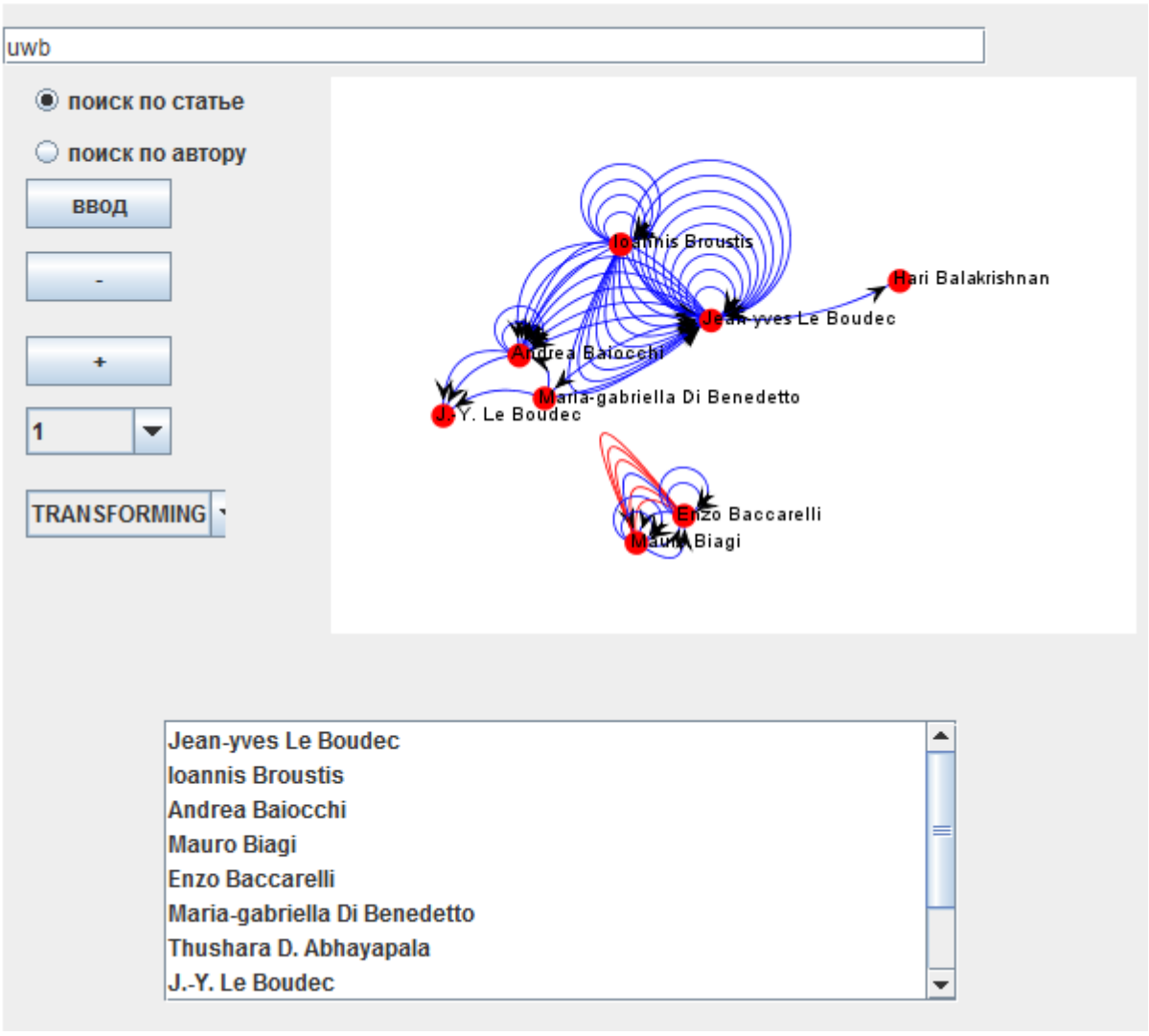


*Рисунок 3*

Отсюда можно выделить две маленьких научных группы, люди внутри которых писали какие-нибудь статьи по теме «uwb».

В одной из тих групп всего два человека: Mauro Biagi и Enzo Baccarelli, которые не только ссылались на статьи друг друга, но и написали несколько статьей в соавторстве. Поэтому изучая тему uwb, выйдя на информацию о Biagi, можно посмотреть информацию и о Baccerelli и общаться с обоими авторами.

В другой группе больше людей. Из них можно выделить Jean-yves Le Boudec, который, как видно из примера, написал много статей по этой теме, на которые ссылается много других авторов (количество синих ребер, входящих в его вершину)

**Пример2.** Второй пример на рисунке 4 показывает часть научных сообществ по теме «real time system»

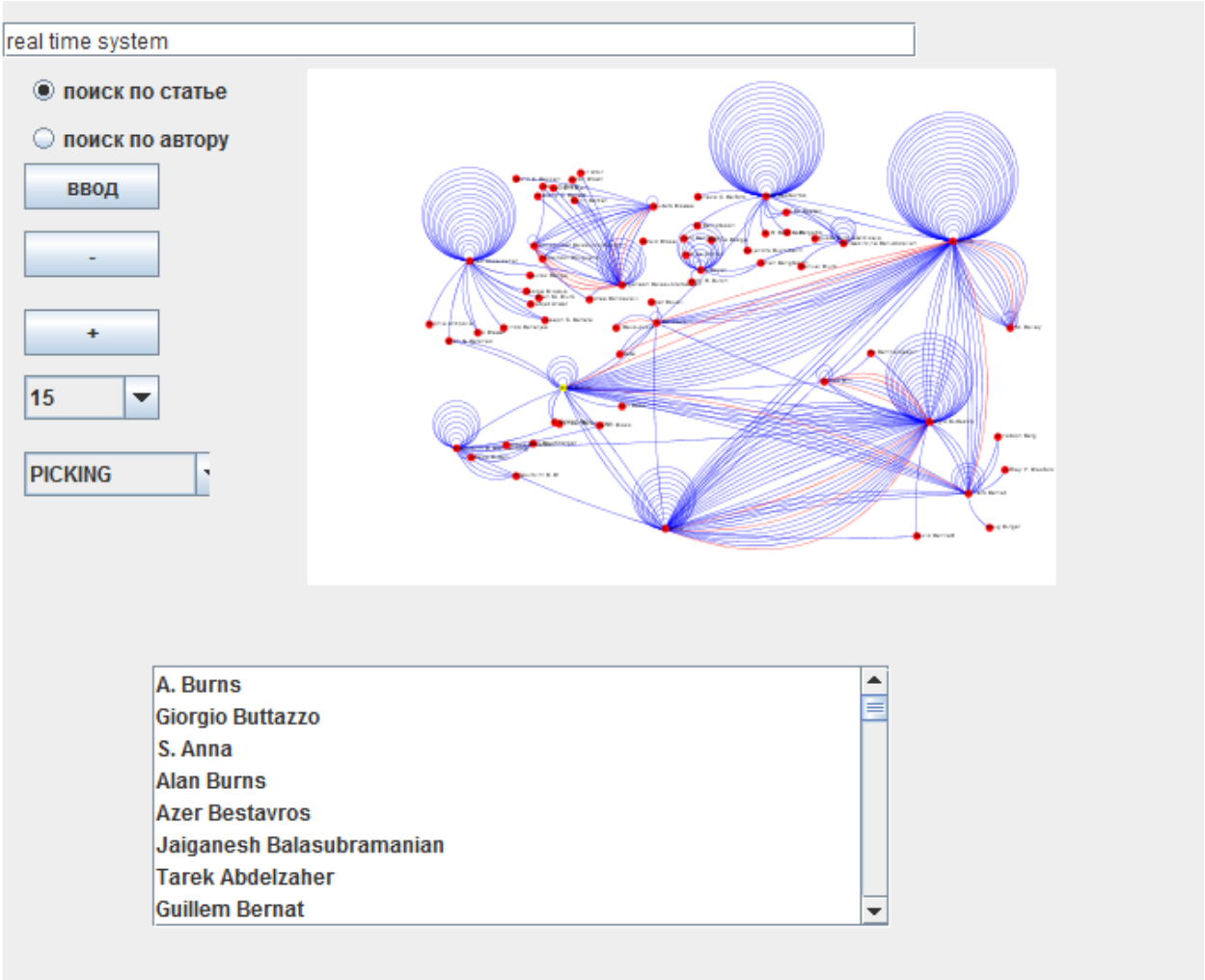


*Рисунок 4*

Данный пример не заслуживает подробного обсуждения, так как из рисунка видно, что большое число авторов написали статьи по теме «real time system», и многие из них заслуживают уважения при обсуждении данной темы.

# Список используемой литературы


1. http://ru.wikipedia.org/wiki/Data_mining - информация о Data Mining
2. http://jung.sourceforge.net/ - информация о библиотеке Jung
3. citeseerx.ist.psu.edu – цифровая библиотек CiteSeer
4. Community Mining Tool using Bibliography Data, Ryutaro Ichis e, Hideaki Takeda National Institute of Informatics 2-1-2 Hitotsubashi Chiyoda-ku Tokyo, 101-8430, Japan – статья об японской разработке

5. http://www.mysql.ru/ - информация о MySql
6. dl.acm.org – информация о цифровой библиотеке ACM
7. http://ci.nii.ac.jp/ - информация о цифровой библиотеке CiNii
8. Aaron Clauset, M. E. J. Newman, and Cristopher Moore. Finding community structure in very large networks, 2004.
9. http://en.wikipedia.org/wiki/Social_graph