AMiner-mini: A People Search Engine for University

Jingyuan Liu*, Debing Liu*, Xingyu Yan*, Li Dong[#], Yutao Zhang*, and Jie Tang*
*Department of Computer Science and Technology, Tsinghua University
[#]Tsinghua University Library, Tsinghua University
{toothacher17,zhangyutao1106}@gmail.com, {jietang,debingliu}@tsinghua.edu.cn, dongli@lib.tsinghua.edu.cn

ABSTRACT

We present a distributed academic search and mining system—AMiner-mini. The system offers intra- and inter- university level academic search and mining services. It integrates academic data from multiple sources and performs disambiguation for people names, which is a fundamental issue for searching people. We employ a two-phases approach that formalizes the disambiguation problem into a probabilistic HMRF framework, which significantly improves the disambiguation performance. Based on the disambiguation results, AMiner-mini offers a people search function, which returns experts (or related researchers) for a given query by the user. The user can also choose different metrics to rank the search results and explore the results from different dimensions. The system is designed in a distributed structure. It can be deployed in a university as a stand-alone system for finding the right people who are working on a research topic. Multiple distributed systems can be also connected via Web services and perform search or mining in an asynchronous way and return the combination results. We have deployed the system in Tsinghua University and feedback from university academic users shows that the system worked well and achieved its primary objective.

Categories and Subject Descriptors

H.3.3 [**Information System**]: Information Search and Retrieval; H.2.8 [**Database Management**]: Database Applications

General Terms

Algorithms, Experimentation

Keywords

Name Disambiguation, Academic Search, Distributed System.

1. INTRODUCTION

With the rapid proliferation of digital academic information, it is becoming more and more challenging for mining the heterogeneous academic knowledge in order to satisfy different usage scenarios like expert finding [2,3] and academic search [5]. For example, in a university, students may want to find the best advisors to work with; faculties are trying to look for the best collaborators from different research fields. Traditional keyword based document search is clearly far from sufficient to meet these requirements.

Traditional digital library system [4] is mainly designed to manage the *digital data*, e.g., to to collect digital information, build index and offer retrieval services for users. However, the trend of Web turns to be more *people-centric* rather than *data-centric*. Vivo System is a university-oriented system led by Cornell and shares motivations as mentioned before. However, it is also not in particular designed for searching people.

In this work, we present AMiner-mini system, a people search engine for university. The system is derived from ArnetMiner [5], but with more people-centric feature and with a distributed structure. The system has two *distinct advantages*: First of all, it can easily incorporate data inside a university (e.g., the library data and the faculty information), which not only offers a way to seamlessly integrate with existing systems, but also be able to use those data to improve the performance of name disambiguation; Second, we design and implement AMiner-mini as distributed system so that it can easily deploy to and connect other universities, which enables the system to conduct inter-university distributed search [4]. System's major contributions can be concluded as follows:

- *Name Disambiguation*: System employs a two-phases name disambiguation approach via integrating department and faculty staff information into a HMRF framework;
- *Academic Search:* We consider three factors, including relevance, importance, and popularity, when designing ranking algorithms for intra-university academic search;
- *Distributed Structure:* We propose distributed structure and mainly study re-ranking algorithms considering importance and serendipity for inter-university distributed search.

In the following sections, we first introduce the system architecture and then explain the core technologies used in the system. Finally, we give the demonstration plan.

2. TECHNOLOGY SPECIFICATION

2.1 System Architecture

AMiner-mini is a people search engine for univeristies and it is designed on a distributed platform. Figure 1 shows system architecture, with each node representing a single university.

The system mainly consists of the following components:

- *Data Preparation:* In this component, we preprocess data for the following search and mining. We extracted data from university library, complete missing data by extracting information from the public Web by automatic information extraction [5]. We also provide an interface for users to edit the potential incorrect extraction. Finally, all data has been stored in an MySQL database.

CIKM'14, Nov 03–07 2014, Shanghai, China.
ACM 978-1-4503-2598-1/14/11.
http://dx.doi.org/10.1145/2661829.2661852

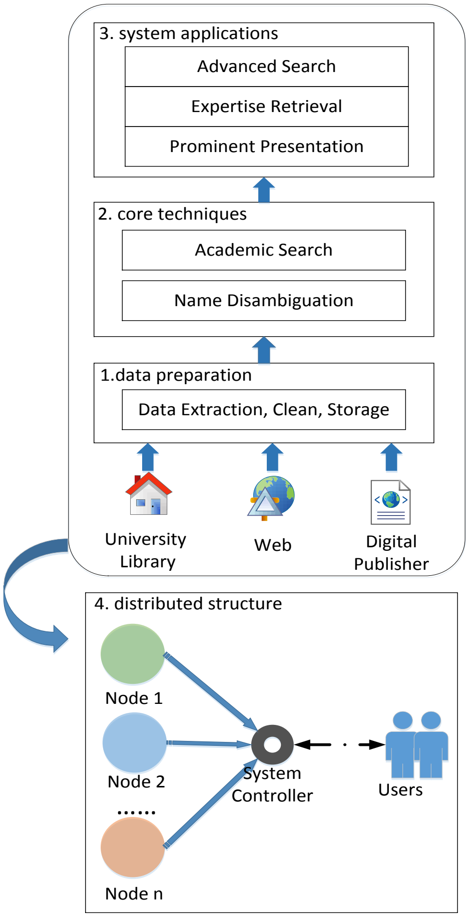


Figure 1: Architecture of AMiner-mini

- *Core Techniques:* This component is our major technical contribution. After data preparation, we design and test our algorithms including two-phases Name Disambiguation and intra-university Academic Search [3]. The former one solves name ambiguity problem when assigning papers and courses to faculties. The latter one is used to measure ranking scores for entities given a user query.
- *System Applications:* We implement system applications based on core techniques in this component. *Prominent Presentation* is used to present the prominent faculties with specific honorable titles, which is attractive to users like administrative officials. Expertise Retrieval [2] is to find expertise entities including faculties, courses, and papers in system database given a search query. Advanced Search is to search with several specific filtering requirements.
- *Distributed Structure:* AMiner-mini is designed with a distributed structure with each node representing a university [4]. On the top of the distributed structure, we studied re-ranking algorithms for Inter-university Distributed Search.

2.2 Name Disambiguation

Different faculties may share identical names. The results in the name ambiguity problem. Name ambiguity will greatly hurt the quality of most academic search and mining services. Thus we tackle name disambiguation as the first step. Name disambiguation generally includes two challenging subtasks: (a) how to cluster information of one person together; and (b) how to determine the number of persons who have the same name [1]. In AMiner-mini, we use a two-phases approach to solve these challenges. In the approach, we leverage human labeled data as supervised constraints to help improves clustering performances.

More specifically, we cooperated with university library and accessed official human-labeled entities of different departments, which is a strong constraint that helps clustering and assignment. Disambiguation scale is set in a department range and we assigned entities (different kinds of information) to faculties within the department. Official labeled data is limited, but it can greatly improve algorithm performances via working as primarily assigned information.

Further, we integrate the above result into a probabilistic HMRF framework [1][5]. We first enrich academic knowledge from extern digital publisher, which does not have department information as constrains. We heuristically define paper attributes, paper relations and author attributes from first phase result as features for HMRF objective function [5]. During assignment iterations [1], we cluster primarily assigned papers together with the unassigned. The primarily assigned information is helpful considering it can "propagate" during iteration. Besides, the primarily assigned information could work as "cluster atom" to improve clustering performances [1]. We employ supervised learning methods such as Naïve Bayes, and SVM to train a classification model for improving the clustering performance [6].

2.3 Academic Search

Academic Search is to find expertise entities (people and other entities like courses) given a query and the key is to measure ranking scores [3]. In AMiner-mini, we studied intra-university academic search. The ranking score is:

$$Score_{intra} = \omega_R * Relevance + \omega_I * Importance_{intra} + \omega_p * Popularity_{intra} \qquad (1)$$

where *Relevance*, *Importance*$_{intra}$, and *Popularity*$_{intra}$ are the three factors, ω are their weights and set as 0.6, 0.2, 0.2 separately.

Relevance. *Relevance* is used to measure relevance between queries and entities. Given a query, for example "data mining", users not only want to find entities containing those words, but also intend to search for entities on "data mining" topics [5]. Blei et al. introduces LDA, an effective topic model for text and has been applied to academic search [3].

However, LDA alone is usually considered as "general" but "not specific", so "coarse" for search [3]. To balance between "generality" and "specificity", we combined LDA with LM.

Importance$_{intra}$. *Importance*$_{intra}$ is used to measure intra-university entity importance. We define "prominent importance" to distinguish the prominent in a university. We heuristically assign higher grades to an entity if it enjoys some titles defined as important such as "fellow of Chinese Academy of Science".

We also consider the "network importance". We build social network from co-authorship of faculties and use random walk with restart to rank entities [3].

Popularity$_{intra}$. *Popularity*$_{intra}$ is used to measure intra-university entity popularity. System logs user behaviors and calculate "feedback popularity" to better understand and measure their preferences [7]. For example, if most users click on the third entity in search result, we would treat the third one as more popular than the former two.

Current system collects 10,000+ faculties, 40,000+ courses, and 90,000+ papers in *Tsinghua University*. System has been put into operation since early 2014. At the beginning, *Popularity*$_{intra}$ is initialized with the same weight for all entities. We test different weights of LDA and LM. Regarding baselines, we use the TFIDF method implemented in Lucene, a popular free software for indexing and search full-text. We test 90 queries and asked 5 computer science majors (2 undergraduates, 1 phD students and 2 engineers) to label the search result.

Table 1. LDA + LM combination weights experiments

Search Methods	**P@5**	**P@10**	**MAP**
0.3 LDA + 0.7 LM	**0.876**	0.8	**0.912**
0.2 LDA + 0.8 LM	0.864	**0.81**	0.89
0.0 LDA + 1.0 LM	0.872	0.77	0.79
Lucene (TFIDF)	0.773	0.726	0.73

As shown in the table, we can see the combination model obviously outperforms Lucene in terms of P@5, P@10 (Precision at top 5 and 10), and also MAP. Regarding the weight configuration for combining LDA and LM, 0.3 for LDA and 0.7 for LM achieves the best performance.

2.4 Distributed Structure

AMiner-mini is designed with a distributed architecture. It can deploy to multiple different universities and the search function can be connected each other. For Cohesion, every single university is considered as a node can works alone. For Concurrency, every node could concurrently react to the system and fasten the responding speed [8].

With the distributed structure, the system is able to offer inter-university distributed search [4]. Users search a query and the master server sends it to all other server nodes. All nodes concurrently conduct academic search and report back the result. The controller collects all search results then re-ranks it. The re-ranking score is:

$$Score_{inter} = \omega_R * Relevance + \omega_I * Importance_{inter} + \omega_p * Popularity_{inter} \qquad (2)$$

where $Importance_{inter}$ denotes the inter-university importance and $Popularity_{inter}$ denotes the inter-university popularity, which are modified based on intra-university academic search.

***Importance*$_{inter}$.** $Importance_{inter}$ is used to measure inter-university entity importance. Entity importance from different universities varies. For example, users may view faculties from *Tsinghua* and *BUPT* differently. The challenge is how to quantitatively measure the "University importance". We can employ public school ranking list and user log statistic to initially calculate the score.

***Popularity*$_{inter}$.** $Popularity_{inter}$ is used to measure inter-university entity popularity. With the distributed structure, search results are largely enriched and simply use "feedback popularity" may cause search dilemma. For example, *A* is ranked before *B* and they are both very accurate for a search query. However, users may seldom notice *B* due to search page settings and increase *A*'s popularity score unfairly, which somehow just violates the real meaning of popularity: to let the users define the most popular entities.

We define "serendipity popularity". Serendipity concern with the novelty of inter-university search result [9] and the challenge is how to assign "serendipity popularity" score fairly and effectively. Heuristically, we randomly set "serendipity popularity" scores. The overall $Popularity_{inter}$ is as follows:

$$Popularity_{inter} = \theta_p * Popularity_{intra} + \theta_S * Serendipity \qquad (3)$$

where $Popularity_{intra}$ is the intra-university popularity score and *Serendipity* is the random serendipity score in a reasonable range, θ is used to normalize scores to the same scale.

3. DEMONSTRATION PLAN

We will present AMiner-mini in the following aspects:

- First, we will use a poster to give an overview of system architecture and briefly show system applications.
- Next, we will describe system's core techniques including name disambiguation and academic search in details.
- After that, we will introduce proposed distributed structure and inter-university search under it. The audience will gain more deep understanding on re-ranking algorithm.
- Finally, we will share our thoughts on the strengths and weakness of system. We will further discuss future work.

4. ACKNOWLEDGEMENTS

The work is supported by the National High-tech R&D Program (No. 2014AA015103), National Basic Research Program of China (No. 2014CB340500), Natural Science Foundation of China (No. 61222212), and Beijing key lab of networked multimedia. We also would like to thank Tsinghua University Library for official data and open test platform.

5. REFERENCES

[1] J. Tang, A.C.M. Fong, B, Wang, and J. Zhang. A Unified Probabilistic Framework for Name Disambiguation in digital library. In *TKDE*, Volume 24, Issue 6, Pages 975-987, 2012

[2] K. Balog, Y. Fang, M. de Rijke, P. Serdyukov and L. Si. Expertise Retrieval. In *FTIR,* Volume 6, 2012

[3] J. Tang, J. Zhang, R. Jin, Z. Yang, K. Cai, L. Zhang, and Z. Su. Topic Level Expertise Search over Heterogeneous Networks. In *Machine Learning Journal*, Volume 82, Issue 2, Pages 211-237, 2011

[4] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval (2nd Edition)*. China Machine Press, 2010

[5] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang and Z. Su. ArnetMiner: Extraction and Mining of Academic Social Network. In *KDD'08*, pages 990-998, 2008.

[6] A. Ferrreira, M. Gnocalves, and A. Laender. A Brief Survey of Automatic Methods for Author Name Disambiguation. In *SIGMOD'12*, 2012

[7] T. Joachims, L. Granka, H. Hembrooke, F. Radlinski, and G. Gay. Evaluating the Accuracy of Implicit Feedback from Clicks and Query Reformulations in Web Search. In *TIS,* Volume 25, 2007

[8] G. Coulouris, J. Dollimore, and T. Kindberg. *Distributed systems: Concepts and Design (5th Edition)*. China Machine Press, 2011.

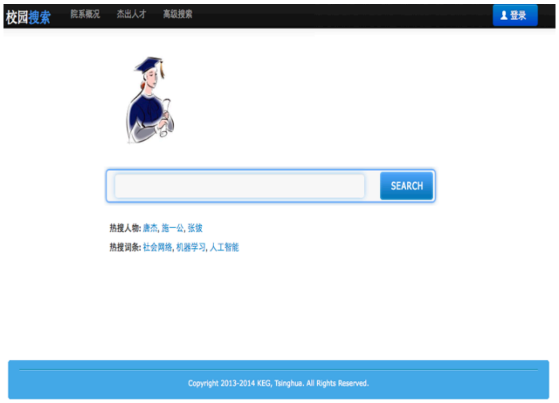
[9] M. Ge, C. Delgado-Battenfeld, and D. Jannach. Beyond accuracy: Evaluating recommender systems by coverage and serendipity. In *RecSys'10*, 2010

6. APPENDIX

Online system URL is: http://dlib.lib.tsinghua.edu.cn. Please note that AMiner-mini is an ongoing project. Visitors should expect the system to change.

System Home Page

All system functions can be accessed from home page. The system homepage is as follows:



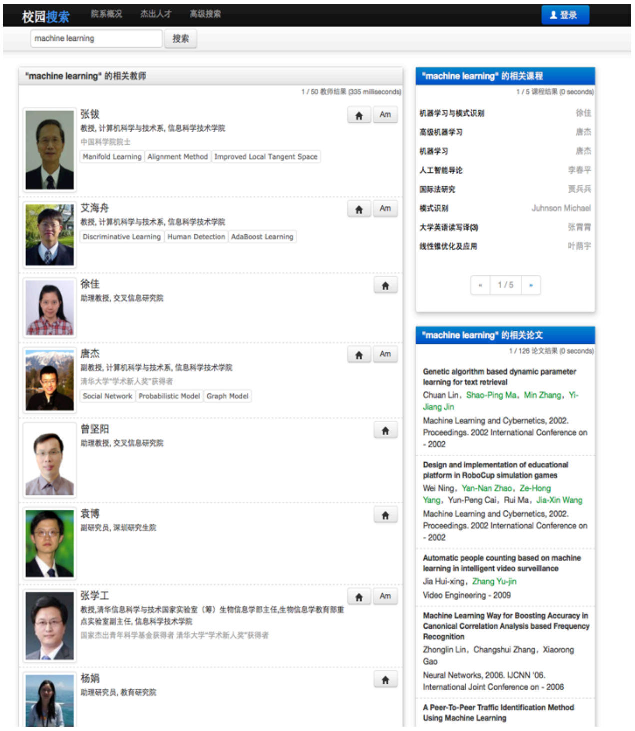
热搜人物: 唐杰, 施一公, 张钹

热搜词条: 社会网络, 机器学习, 人工智能

Copyright 2013-2014 KEG, Tsinghua. All Rights Reserved.

Academic Search

When the user inputs a query in the search box, the system return a search result consisting of faculties, publications, and courses. For example, searching "machine learning", the result is as follows:



machine learning

"machine learning" 的相关教师

1 / 50 教师结果 (335 milliseconds)

张钹
教授, 计算机科学与技术系, 信息科学技术学院
中国科学院院士
Manifold Learning | Alignment Method | Improved Local Tangent Space

艾海舟
教授, 计算机科学与技术系, 信息科学技术学院
Discriminative Learning | Human Detection | AdaBoost Learning

徐佳
助理教授, 交叉信息研究院

唐杰
副教授, 计算机科学与技术系, 信息科学技术学院
清华大学"学术新人奖"获得者
Social Network | Probabilistic Model | Graph Model

曾坚阳
助理教授, 交叉信息研究院

袁博
副研究员, 深圳研究生院

张学工
教授,清华信息科学与技术国家实验室（筹）生物信息学部主任,生物信息学教育部重点实验室副主任, 信息科学技术学院
国家杰出青年科学基金获得者 清华大学"学术新人奖"获得者

杨娟
助理研究员, 教育研究院

"machine learning" 的相关课程

1 / 5 课程结果 (0 seconds)

机器学习与模式识别	徐佳
高级机器学习	唐杰
机器学习	唐杰
人工智能导论	李春平
国际法研究	贾兵兵
模式识别	Juhnson Michael
大学英语读写译(3)	张霄霄
线性锥优化及应用	叶荫宇

« 1 / 5 »

"machine learning" 的相关论文

1 / 126 论文结果 (0 seconds)

Genetic algorithm based dynamic parameter learning for text retrieval
Chuan Lin, Shao-Ping Ma, Min Zhang, Yi-Jiang Jin
Machine Learning and Cybernetics, 2002. Proceedings. 2002 International Conference on - 2002

Design and implementation of educational platform in RoboCup simulation games
Wei Ning, Yan-Nan Zhao, Ze-Hong Yang, Yun-Peng Cai, Rui Ma, Jia-Xin Wang
Machine Learning and Cybernetics, 2002. Proceedings. 2002 International Conference on - 2002

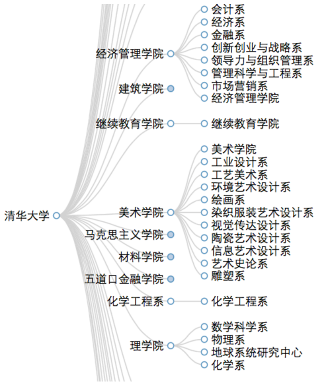
Automatic people counting based on machine learning in intelligent video surveillance
Jia Hui-xing, Zhang Yu-jin
Video Engineering - 2009

Machine Learning Way for Boosting Accuracy in Canonical Correlation Analysis based Frequency Recognition
Zhonglin Lin, Changshui Zhang, Xiaorong Gao
Neural Networks, 2006. IJCNN '06. International Joint Conference on - 2006

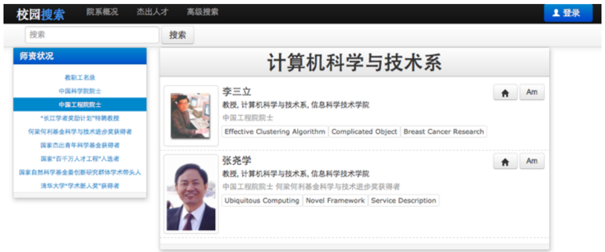
A Peer-To-Peer Traffic Identification Method Using Machine Learning

Department View

We use "Collapsible Trees" from D3.js to present department information as follows:



When choosing a certain department, users can view faculty information organized by faculty titles in the department, for example, Department of Computer Science and Technology:



师资状况

- 教职工名录
- 中国科学院院士
- 中国工程院院士
- "长江学者奖励计划"特聘教授
- 何梁何利基金科学与技术进步奖获得者
- 国家杰出青年科学基金获得者
- 国家"百千万人才工程"入选者
- 国家自然科学基金委创新研究群体学术带头人
- 清华大学"学术新人奖"获得者

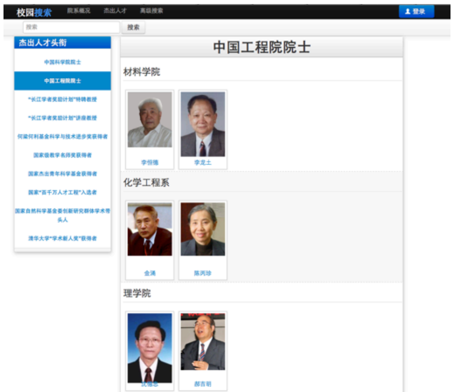
计算机科学与技术系

李三立
教授, 计算机科学与技术系, 信息科学技术学院
中国工程院院士
Effective Clustering Algorithm | Complicated Object | Breast Cancer Research

张尧学
教授, 计算机科学与技术系, 信息科学技术学院
中国工程院院士 何梁何利基金科学与技术进步奖获得者
Ubiquitous Computing | Novel Framework | Service Description

Prominent Presentation

Prominent presentation shows faculties which honorable titles. The Prominent View is as follows:



杰出人才头衔

- 中国科学院院士
- 中国工程院院士
- "长江学者奖励计划"特聘教授
- "长江学者奖励计划"讲座教授
- 何梁何利基金科学与技术进步奖获得者
- 国家级教学名师奖获得者
- 国家杰出青年科学基金获得者
- 国家"百千万人才工程"入选者
- 国家自然科学基金委创新研究群体学术带头人
- 清华大学"学术新人奖"获得者

中国工程院院士

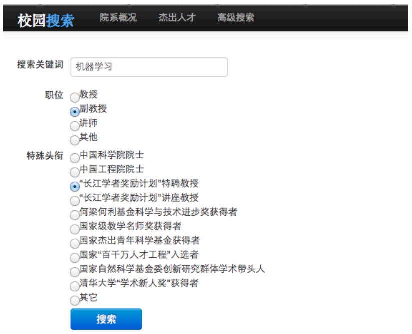
材料学院
李恒德 李龙土

化学工程系
金涌 陈丙珍

理学院
沈德忠 郝吉明

Advanced Search

We offer advanced search services based on the academic search. In advanced search, users can search entities with specific filters.



搜索关键词 机器学习

职位
○ 教授
● 副教授
○ 讲师
○ 其他

特殊头衔
○ 中国科学院院士
○ 中国工程院院士
● "长江学者奖励计划"特聘教授
○ "长江学者奖励计划"讲座教授
○ 何梁何利基金科学与技术进步奖获得者
○ 国家级教学名师奖获得者
○ 国家杰出青年科学基金获得者
○ 国家"百千万人才工程"入选者
○ 国家自然科学基金委创新研究群体学术带头人
○ 清华大学"学术新人奖"获得者
○ 其它

搜索