夫裴特卡罗法(MCMC)、交替最小二乘法(ALS).其中,SGD 最为常用且具有较快的训练速度,本文的参数求解也使用该方法.

4 实验设计

4.1 数据集选取与实验设置

数据集采用从 ProgrammableWeb 平台上爬取的真实数据,包含 6 673 个 Mashup,9 121 个 Web APIs 以及 13 613 个 Web API 和 Mashup 之间的链接.由于 HDP 模型主要依据词频共现来聚类分组数据,为了导出最相关的主题,我们对 Web APIs 和 Mashup 的描述文档做预处理.图 5 给出了名称为“Earthmine Flash Viewer”的 Web API 描述文档预处理的详细过程.

Web API 名称	Earthmine Flash Viewer
第 1 步:选取 Mashup 和 Web APIs 的描述文档作为原始文件,每一行代表一个 Mashup 或 Web API.	Integrate street-level images into your site via this Flash Viewer. Each image has longitude and latitude data, and you can do geo searches, keyword searches, and more. Contact for more information. Like the earthmine Flash Viewer API? Click the “Track this API” button on any profile page and never miss an API update, new app, or breaking news for that API again.
第 2 步:去除不合理的特征(包括,.;:=+?@%$|等).	Integrate street level images into your site via this Flash Viewer Each image has longitude and latitude data and you can do geo searches keyword searches and more Contact for more information Like the earthmine Flash Viewer API Click the Track this API button on any profile page and never miss an API update new app or breaking news for that API again
第 3 步:将所有字母均转换为小写字母.	Integrate street level images into your site via this flash viewer each image has longitude and latitude data and you can do geo searches keyword searches and more contact for more information like the earthmine flash viewer api click the track this api button on any profile page and never miss an API update new app or breaking news for that api again
第 4 步:去除停止词,并使用英语词典校验单词.	Integr street level imag site flash viewer imag ha longitud latitud data geo search keyword search contact inform earthmin flash viewer api click track api button ani profil api updat app break api
第 5 步:去除超高频且不具区分度的词(service,API)、词频为 1 的词.此类词既不具有区分度,且在训练语料库是严重消耗系统资源.	Integr street level imag site flash viewer imag longitud latitud geo search keyword search contact inform earthmin flash viewer click track button ani profil updat app break
第 6 步:使用 POSTagger 工具标注单词词性.	Integr/JJ street/NN level/NN imag/NN site/NN flash/VBP viewer/NN imag/NN longitud/NN latitud/JJ geo/NN search/NN keyword/NN search/NN contact/NN inform/VBP earthmin/JJ flash/NN viewer/NN click/VBP track/NN button/NN ani/NN profil/NN updat/NN app/NN break/NN
第 7 步:选取名词词性及动词词性的单词作为最终训练的语料库.	Street level imag site flash viewer imag longitud geo search keyword search contact inform flash viewer click track button ani profil updat app break
HDP 算法的输入文件格式(每一行为一个词频向量):描述文档词数 第 1 个词索引:第 1 个词的词频 第 2 个词索引:第 2 个词的词频...	24 533:149 266:319 57:1307 1056:59 444:186 930:69 266:319 459:180 519:152 10:3788 180:473 10:3788 109:770 9:4266 444:186 930:69 76:1005 35:1916 120:711 39:1837 61:1240 52:1514 4268:6 134:667

Fig.5 A case analysis of experimental pretreatment

图 5 实验预处理案例分析

- 实验环境设置:利用 Java 语言实现 HDP 算法,运行在一台 60G 内存的服务器;
- 实验中参数设置:HDP 算法迭代次数 Iter 为 3 000,HDP 模型参数 α=1, η=0.1, γ=1.5.实验中,Top-A 个 APIs 和 Top-M 个 Mashup 在因子分解机模型中分别设置为 10 和 20.实验中,Mashup 数据集被随机平均分成 5 个部分,其中一个部分为测试集,而另外 4 个部分边为训练集.

4.2 实验的评测指标

本文采用召回率、准确率、*F*-measure 和 *NDCG@N* 指标来评价以上 6 种方法性能的优劣,分别定义如下.

- 召回率反映的是推荐的相关 Web APIs 占所有相关 Web APIs 的比率,计算如公式(13)所示.

$$\text{召回率}=\frac{|R(A_i)\cap RM(A_i)|}{RM(A_i)} \tag{13}$$

- 准确率反映推荐的 Web APIs 集合中相关 Web APIs 所占的比例,计算如公式(14)所示.

$$准确率=\frac{|R(A_i)\cap RM(A_i)|}{R(A_i)} \tag{14}$$

- *F*-measure 是召回率与准确率的调和平均值,即

$$F\text{-measure}=\frac{2\times 召回率\times 准确率}{召回率+准确率} \tag{15}$$

其中,$R(A_i)$表示相关的 Web APIs(被目标 Mashup 真实调用的 Web APIs),$RM(A_i)$表示推荐的 Web APIs.

- *NDCG@N*(normalized discounted cumulative gain:归一化折损累积增益).在信息检索领域中,该方法是一种流行的的衡量排序质量的指标.本文用来衡量推荐列表中推荐 Web APIs 排名的优劣.*NDCG@N* 的值越高,说明 Web APIs 的推荐列表排序结果越好.即

$$DCG@N=\sum_1^N\frac{2^{rel_i}-1}{\log_2(1+i)} \tag{16}$$

$$NDCG@N=\frac{DCG@N}{IDCG} \tag{17}$$

其中,

- N 表示 Web APIs 的推荐个数;
- rel_i 表示第 i 个推荐的 Web API 的相关性得分:如果推荐的第 i 个 Web API 就是真实数据集中 Mashup 调用的 Web APIs,此时 rel_i=1;否则,rel_i=0;
- *IDCG*(ideal *DCG@N*),就是最大的 *DCG@N* 值(*DCG@N* 可以通过公式(16)计算得到),即为最优的推荐情况.

4.3 比较方法

- TF-IDF[1]:

利用词向量空间模型推荐与目标 Mashup 描述文档相似的 Web APIs.假设目标 Mashup m 的词向量空间为 $V^{(m)}$,第 i 个 Web API 的词向量空间为 $V^{(a_i)}$,则目标 Mashup m 与第 i 个 Web API 的文本相似度计算如公式(18)所示.

$$Sim(m,a_i)=\frac{V^{(m)}V^{(a_i)}}{\|V^{(m)}\|\|V^{(a_i)}\|} \tag{18}$$

最后,该 Web API 预测评分通过公式(19)计算得出.

$$Score(m,a_i)=pop(a_i)Sim(m,a_i) \tag{19}$$

其中,$pop(a_i)$为第 i 个 Web API 的流行度,可通过第 3.2 节中的公式(7)计算.

- E-LDA[3]

该方法使用 LDA 模型分别导出的 Mashup 和 Web API 的主题分布向量;接着,利用增强余弦相似度公式(6)来度量 Mashup 与 Web APIs 之间的文本相似度 $S(m,a_i)$;最后,将相似度高且流行的 Top-N 个 Web APIs 推荐给目标 Mashup.Web API 预测评分如公式(20)所示.

$$Score(m,a_i)=pop(a_i)S(m,a_i) \tag{20}$$

- E-HDP

类似于 E-LDA 推荐方法,该方法推荐相似度高且流行的 Top-N 个 Web APIs 给目标 Mashup.不同的是,该方法获取服务描述文档主题分布向量是由 HDP 模型导出.

- LDA-CF[6]

利用 LDA 模型与协同过滤方法共同获取候选的 Web API 集合,推荐候选集合中最流行的 Top-N 个 Web APIs 供 Mashup 创建使用.公式(21)给出了相应的推荐 Web APIs 的获取.

$$RecommendAPI = pop\text{Max}_N\{Cond_{i=i}^{I}(m,a_i)\} \tag{21}$$

其中, $Cond_{i=i}^{I}(m,a_i)$ 为 Mashup m 的候选 Web API 集合,$popp\text{Max}_N\{\cdot\}$表示取流行度最高的前 N 个 Web APIs.

- HDP-CF

类似于 LDA-MF 推荐方法的实现过程,在此基础上,使用 HDP 模型代替 LDA 模型训练服务描述文档的主题分布向量.

- LDA-FM

利用因子分解模型的优势,同时考虑文本相似度、Web API 的流行度与 Web API 的共线性,预测评分推荐 Top-N 个得分最高的 Web APIs 辅助 Mashup 创建.

- HDP-FM

利用因子分解模型融合文本相似度、Web API 的流行度、Web API 的共线性,预测评分推荐 Top-N 个得分最高的 Web APIs 辅助 Mashup 创建.与 LDA-FM 方法唯一的不同是,服务描述文本的主题分布向量由 HDP 模型训练获得.

4.4 实验评估

本小节我们首先对 HDP 模型的训练过程进行观测,接着探讨了主题 K 的选取对实验结果的影响,最后选取准确率、召回率、F-measure 和 $NDCG@N$ 这 4 种评价指标对多种方法进行比较.

(1) HDP 模型训练观测

HDP 模型在训练结束后会获得训练语料库的最优主题个数、每个主题下的词分布以及每一个 Mashup 和 Web API 的主题分布概率.表 2 展示了利用 HDP 模型获取的部分“主题-词”分布的情况,如,sport game score player 等词被分配到 Topic 14 中.表 3 展示了部分 Web API 描述文档在表 3 所给出的主题下的分布情况.如,Web API “FishingBuddy”在 Topic 14 上的分布率为 0.324059.表 4 给出了 HDP 模型的α,η,γ参数与主题数 K 随迭代次数变化的部分记录情况,不难发现,当迭代次数约达到 2 172 次时,主题个数不再发生变化,迭代过程趋于稳定,此时获取最优主题数 K=76.

Table 2 Some topics and their representative words learned from HDP

表 2 利用 HDP 模型获取的部分“主题-词”分布

主题	该主题下的词
Topic 14	sport game score player statistic football team new
Topic 26	call voice audio phone chat telephony record conference meet response
Topic 39	communication text language tool analysis extraction content word sentiment dictionary search
Topic 48	travel estate hotel restaurant deals offer website airport trip flight
Topic 70	location map weather latitude longitude area measurement

Table 3 Some topic distribution of Web APIs document

表 3 部分 Web API 描述文档的“文档-主题”分布

Web API 名称	Topic 14	Topic 26	Topic 39	Topic 48	Topic 70
FishingBuddy	0.324 059	0.018 313	0.013 059	0	0.012 987
360voice	0.318 004	0.016 112	0.009 914	0.007 115	0.012 545
iLime	0.015 484	0.321 454	0	0.155 458	0.001 245
eKlima	0	0.081 231	0.004 611	0.035 529	0.422 245
SharedBookshelve	0.012 454	0.012 154	0.015 54	0.017 454	0.013 254

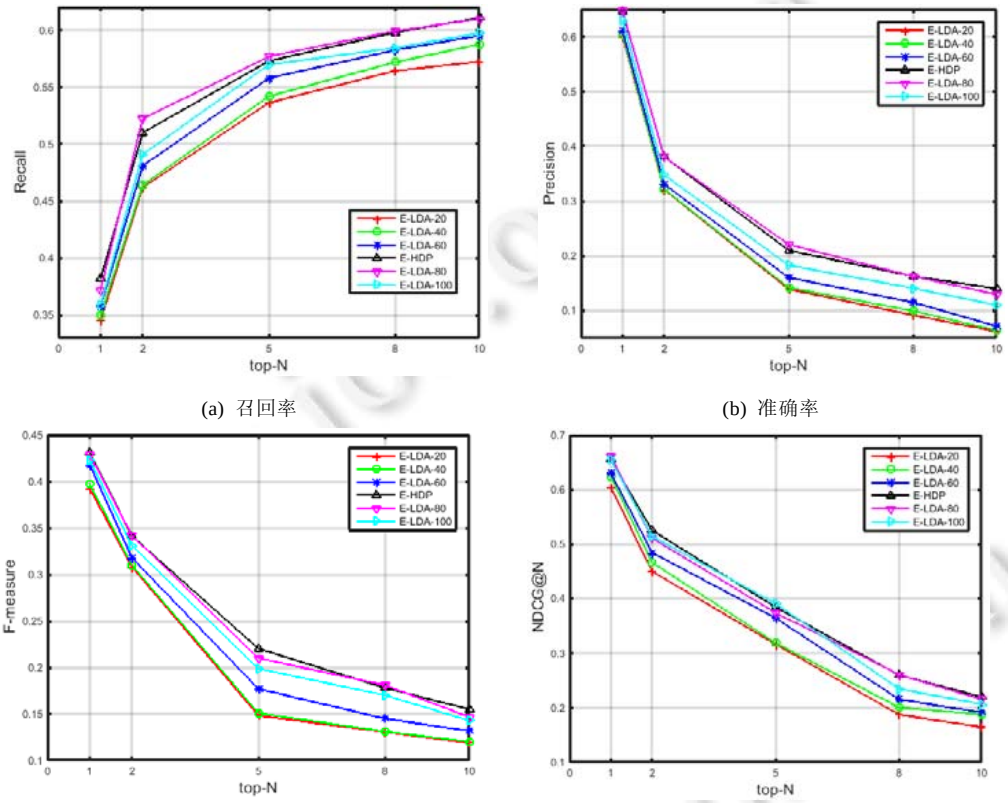
Table 4 Sample iteration record of parameters and the number of topic

表 4 参数与主题数迭代记录(部分)

迭代次数	K	α	η	γ
1	13	1.000 00	0.100 00	1.500 00
100	14	0.385 67	0.002 62	2.440 67
1 000	42	0.555 12	0.000 92	10.874 17
2 000	73	0.812 31	0.000 56	25.355 29
2 072	76	0.823 83	0.000 54	20.879 48
2 400	76	0.864 07	0.000 52	24.919 24

(2) 主题数 *K* 的影响

本组实验分别将 E-LDA 方法中的主题数 *K* 设置为 20,40,60,80,100 与 E-HDP 方法进行比较,E-HDP 方法的主题数 *K*=76 由 HDP 模型自动生成.图 6 中的纵坐标表示相应的评价指标(准确率、召回率、*F*-measure 及 *NDCG@N*)的值,横坐标为推荐 Web API 的数目,分别设置为 1,2,5,8,10.从图 6 中可以观察到:当主题数 *K* 在 20 至 80 之间时,E-LDA 的性能(准确率、召回率、*F*-measure 及 *NDCG@N* 的值)随着主题数 *K* 的增加而上升;当主题数 *K* 在 80~100 之间时,E-LDA 的性能呈现下降趋势.这说明当主题数 *K* 为 80 时,E-LDA 方法具有最优的性能.从图 6 中还可以发现,E-HDP 方法与主题数 *K* 为 80 时的 E-LDA 方法性能基本相同(准确率、召回率、*F*-measure 及 *NDCG@N* 的曲线相互缠绕近似重叠),证明 HDP 模型能够自动生成最优主题数.



(a) 召回率 (b) 准确率

(c) *F*-measure (d) *NDCG@N*

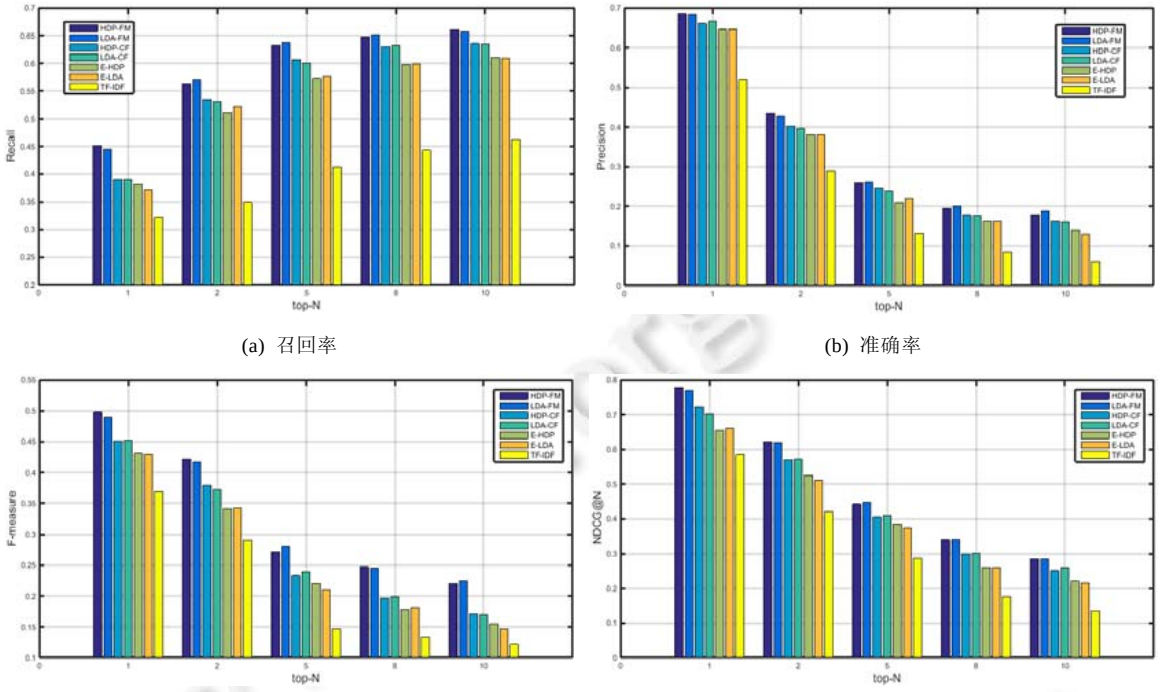
Fig.6 Impact of the number of topic on Web API recommendation

图 6 主题个数选取对 Web API 推荐的影响

(3) 推荐方法比较分析

本组实验将 HDP-FM 与第 4.3 节中介绍的方法进行比较,以此来说明 HDP-FM 具有良好的性能.与第 4.4 节情形(2)中的设置方式类似,图 7 中的纵坐标表示相应的评价指标的值,横坐标为推荐 Web API 的数目.在图 7 中我们不难发现,TF-IDF 方法的性能最差,因为 TF-IDF 方法仅仅利用词向量空间模型度量 Web APIs 与目标 Mashup 之间的相似度,该方法忽略了文档与文档之间的语义信息;其次,E-LDA 及 E-HDP 方法分别引入 LDA 模型和 HDP 模型,这两类主题模型均能挖掘描述文档的潜在语义信息(主题),因此其性能较之 TF-IDF 在相似度的计算上更为精确;再其次,LDA-CF 与 HDP-CF 方法利用协同过滤算法来增强 LDA 及 HDP 算法,以挖掘 Web API 与 Mashup 之间更深层次的链接关系,性能相比仅仅使用 LDA 及 HDP 模型又有所提升.但是 LDA-CF 与 HDP-CF 方法由于协同过滤算法的局限性,无法实现多维度信息的建模,FM 方法能够将多维度的信息作为输入且避免

稀疏性带来的影响,因此,LDA-FM 及 HDP-FM 方法在准确率、召回率、*F*-measure、*NDCG@N* 上表现最优.值得注意的是,实验中,凡是使用到 LDA 模型的方法均通过手动调节选取最优主题数,而涉及 HDP 模型的方法的最优主题数均由 HDP 模型自动生成.从图 7 中我们可以看出,HDP-FM 与 LDA-FM、HDP-CF 与 LDA-CF、E-HDP 与 E-LDA 均具有相似的性能,这进一步说明了 HDP-FM 方法的优越性.



(a) 召回率 (b) 准确率

(c) *F*-measure (d) *NDCG@N*

Fig.7 Recommendation performance comparison of various means

图 7 多种方法推荐的性能比较

5 总结与展望

本文着眼于“根据用户的自然语言描述需求推荐可行的或者推荐可用于解决问题的任务集合”,提出了 HDP-FM 方法用于推荐解决 Mashup 构建问题的 Web APIs 服务集合.HDP-FM 方法通过多维信息处理与多维信息融合两个阶段,将 Mashups 之间的相似度、Web APIs 之间的相似度、Web API 的共现性及流行度输入因子分解机模型,利用评分排序结果,为 Mashup 的创建推荐 Top-*N* Web APIs 作为推荐集合.本文的核心在于“利用 HDP 模型解决最优主题选取问题,利用因子分解机模型解决多维信息融合问题”.为了验证 HDP-FM 方法的有效性,本文分别采用了 4 种评价指标,比较了多种 Web APIs 推荐方法.一系列的实验结果均表明,HDP-FM 算法能够自动确定最优主题,且具有较高的推荐准确性.在未来的工作中,我们将继续探索机器学习技术来进一步提高 Web API 推荐的精度,如,可利用 LSTM(long short-term memory,长短期记忆网络)神经网络算法结合协同过滤、矩阵分解、因子分解机模型进一步提高服务发现的精度.

References:

[1] Cao B, Liu J, Tang M, Zheng Z, Wang G. Mashup service recommendation based on usage history and service network. Int’l Journal of Web Services Research, 2013,10(4):82−101.

[2] Blei DM, Ng AY, Jordan MI. Latent Dirichlet allocation. Journal of Machine Learning Research, 2003,3:993−1022.

[3] Li C, Zhang R, Huai J, Sun H. A novel approach for API recommendation in Mashup development. In: Proc. of the IEEE 21st Int'l Conf. on Web Services. Anchorage, 2014. 289−296.

[4] Chen L, Wang Y, Yu Q, Zheng Z, Wu J. WT-LDA: User tagging augmented LDA for web service clustering. In: Proc. of the Int'l Conf. on Service-Oriented Computing. New York: Springer-Verlag, 2013. 162−176.

[5] Teh YW, Jordan MI, Beal MJ, Blei DM. Sharing clusters among related groups: Hierarchical Dirichlet processes. Advances in Neural Information Processing Systems, 2005,37(2):1385−1392.

[6] Zheng Z, Ma H, Lyu M, King I. Qos-Aware Web service recommendation by collaborative filtering. IEEE Trans. on Services Computing, 2011,4(2):140−152.

[7] Chen X, Zheng Z, Yu Q, Lyu MR. Web service recommendation via exploiting location and QoS information. IEEE Trans. on Parallel Distributed System, 2014,25(7):1913−1924.

[8] Wei L, Yin J, Deng S, Li Y, Wu Z. Collaborative Web service QoS prediction with location-based regularization. In: Proc. of the IEEE 19th Int'l Conf. on Web Services. Honolulu, 2012. 464−471.

[9] He P, Zhu J, Zheng Z, Xu J, Lyu MR. Location-Based hierarchical matrix factorization for web service recommendation. In: Proc. of the IEEE 21st Int'l Conf. on Web Services. Anchorage, 2014. 297−304.

[10] Rendle S. Factorization machines. In: Proc. of the IEEE 10th Int'l Conf. on Data Mining. Sydney, 2010. 995−1000.

[11] Rendle S. Factorization machines with libfm. ACM Trans. on Intelligent Systems and Technology, 2012,3(3):57−78.

[12] Yao L, Sheng QZ, Ngu AHH, Yu J, Segev A. Unified collaborative and content-based web service recommendation. IEEE Trans. on Services Computing, 2015,8(3):453−466.

[13] Cao B, Liu X, Rahman MM, Li B, Liu J, Tang M. Integrated content and network-based service clustering and Web APIs Recommendation for Mashup development. IEEE Trans. on Services Computing, 2017,3(22):1−14.

[14] Gao W, Chen L, Wu J, Gao H. Manifold-Learning based API recommendation for Mashup creation. In: Proc. of the IEEE 22nd Int'l Conf. on Web Services. New York, 2015. 432−439.

[15] Gao W, Chen L, Wu J, Bouguettaya A. Joint modeling users, services, Mashup and topics for service recommendation. In: Proc. of the IEEE 23rd Int'l Conf. on Web Services. San Francisco, 2016. 260−267.

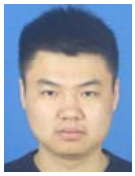
[16] Xia B, Fan Y, Tan W, Huang K, Zhang J, Wu C. Category-Aware API clustering and distributed recommendation for automatic Mashup creation. IEEE Trans. on Services Computing, 2015,8(5):674−687.

[17] Liu X, Fulia I. Incorporating user, topic, and service related latent factors into Web service recommendation. In: Proc. of the IEEE 22nd Int'l Conf. on Web Services. New York, 2015. 185−192.

[18] Pitman J. Combinatorial stochastic processes. Lecture Notes in Mathematics, 2006,1875(94):75−92.

[19] Ishwaran H, James LF. Gibbs sampling methods for stick-breaking priors. General Information, 2003,96(453):161−173.

[20] Abramson N, Braverman D, Sebestyen G. Pattern recognition and machine learning. IEEE Trans. on Information Theory, 2003,9(4): 257−261.



李鸿超(1993－),男,安徽六安人,硕士,CCF 学生会员,主要研究领域为服务计算,服务推荐.

曹步清(1979－),男,博士,副教授,CCF 专业会员,主要研究领域为软件工程,服务计算与云计算.

刘建勋(1970－),男,博士,教授,博士生导师,CCF 杰出会员,主要研究领域为服务计算与云计算,工作流管理的理论与应用.

石敏(1991－),男,软件工程师,CCF 学生会员,主要研究领域为服务计算,数据挖掘,人工智能.