

基于注意力机制与文本信息的用户关系抽取

赵赞，吴璠，王中卿，李寿山，周国栋

(苏州大学 计算机科学与技术学院, 江苏 苏州 215006)

摘要: 随着社交媒体的发展, 用户之间的关系网络对于社交媒体的分析有很大的帮助。因此, 本文主要研究用户好友关系检测。以往的关于用户好友关系抽取的研究主要基于社交媒体上的结构化信息, 比如其他好友关系, 用户的不同属性等。但是, 很多时候用户本身并没有大量的好友信息存在, 同时也不一定有很多确定的属性。因此, 我们希望基于用户发表的文本信息来对用户关系进行预测。不同于以往的潜在好友推荐算法, 本文提出了一种基于注意力机制以及长短时记忆网络 (Long Short-Term Memory, LSTM) 的好友关系预测模型, 将好友之间的评论分开处理, 通过分析用户之间的评论来判断是否具备一定的的好友关系。该模型将好友双方信息拼接后的结果作为输入, 并将注意力机制应用于 LSTM 的输出。实验表明, 用户之间的评论对于好友关系预测确实有较大的实际意义, 本文提出的模型较之于多个基准系统的效果, 取得了明显的提升。在不加入任何其它非文本特征的情况下, 实验结果的准确率达到了 77%。

关键词: 好友判断; 关系预测; 社交网络; 注意力机制

中图分类号: TP391

文献标识码: A

User Relations Extraction via Text Information and Attention Mechanism

ZHAO Yun, WU Fan, WANG Zhongqing, LI Shoushan, ZHOU Guodong

(School of Computer Science & Technology, Soochow University, Suzhou, Jiangsu 215006, China)

Abstract : With the development of social media, the relationship network between users has greatly helped the analysis of social media. Therefore, this paper mainly studies user friend relationship detection. The previous research on the extraction of user friend relationships was mainly based on structured information on social media, such as other friend relationships, different attributes of users, and the like. However, many times the user does not have a lot of friends information, and it does not necessarily have a lot of certain attributes. Therefore, we hope to predict the user relationship based on the text information published by the user. Different from the previous potential friend recommendation algorithm, this paper proposes a friendship prediction model based on attention mechanism and Long Short-Term Memory(LSTM), which separates the comments between friends, and determines whether there is a certain friend relationship by analyzing the user's comments. This model takes as

收稿日期: 定稿日期:

基金项目: 国家自然科学基金 (61331011, 61672366)

作者简介: 赵赞 (1995—), 硕士研究生, 主要研究领域为自然语言处理; 吴璠 (1994—), 硕士研究生, 主要研究领域为自然语言处理; 王中卿 (1987—), 博士, 主要研究领域为自然语言处理, 社会计算。

input the concatenated results of the two friends and applies the attention mechanism to the output of the LSTM. Experiment shows that the comments between users have great practical significance for the prediction of friends relationships. The model presented in this paper has achieved significant improvement over the effects of multiple reference systems. Without adding any other non-text features, the accuracy of the experimental results reached 77%.

Key words: friend judgment; relationship prediction; social network; text categorization

1 引言

社交媒体早已成为互联网一大趋势，以微博以及推特为代表的众多实时交互网络社交应用每日都要产生并传播大量文本信息，其本身构成了一个错综复杂的社交网络。用户之间形成了各种复杂的关系，比如用户之间的关注关系，用户之间的兴趣爱好甚至是偏爱的商家产品等等，这些都可以从这些信息中得到反映。用户之间能够毫不费劲地进行各种实时的互动，其影响力的传播是非常迅速的。因此其信息的传播难以受到实时的控制，其真实性和隐私权难以受到保障。如果能有效地挖掘出这些用户关系，就能极大地遏制住有害信息的传播，对于舆情监控、应对社会突发事件、引导群体行为有着积极的意义。同时还能节省企业的网络营销成本，对用户的个性化推荐也可以起到一定的参考价值。

通常情况下，用户可以视作社交网络拓扑结构中的节点，我们可以通过分析节点之间连接关系从而推断用户之间的关系，也可以直接通过用户标签或行为，比如购买过类似的商品、转发过类似的链接等等来预测可能的关系。但是能够反映用户关系的信息如标签、用户之间的连接等结构化信息往往是有限的，而用户发表的评论则是相当易获得的。因此，本文所描述的好友关系抽取任务就是从用户的文本信息中挖掘出用户之间的潜在好友关系，这不仅仅对产品推荐或是相似好友推荐有相当大的借鉴意义，同时也为通过文本进行人物关系分析有一定的借鉴意义。但是，单靠文本特征判断是有一定的困难的，这不同于直接从用户的某些显著特征或是人物关系链进行判断，而且用户之间的文本数量可能比较巨大，单靠人工判断几乎不可能，但是深度学习却可以充分发挥数据的优势。

一种直接利用文本信息做用户关系抽取的方法是支持向量机（Support Vector Machine, SVM）。但 SVM 无法保留原始文本的顺序信息，这也是词袋模型的一大缺点，而 LSTM 模型能够保留输入的序列特征，因此在本文中主要采用了 LSTM 来构建模型。同时，模型的输入应该要包含好友双方的信息，因此有必要将双方的文本信息进行合理的拼接。此外，注意力机制（Attention）能够使模型关注于好友双方之间更相关的信息。因此，本文比较了两种 LSTM 的拼接方式对于实验结果的影响，并在最终的模型中加入了注意力机制。实验结果表明，在模型输入模型时，直接将好友信息进行拼接的效果最好，加入注意力机制能够明显提升实验准确率。模型最终的实验效果明显超过了传统 SVM。

本文的具体结构如下：第二小节介绍了传统的关系预测和推荐方法以及最近的研究进展，第三小节介绍了数据的收集过程以及数据集相关信息，第四小节详细介绍了我们提出的模型，并简要介绍了特征的处理以及模型理论方面的内容，第五小节介绍了具体的实验设置并比较了多种模型的实验效果，最后是对本文工作的一些总结和展望。

2 相关工作

目前关于好友关系预测的研究主要集中在算法方面，常见的有基于链路预测的算法

和协同过滤算法等。Sarukkai^[1]最开始提出并评估了马尔可夫链进行预测和路径分析。Liben-Nowell 和 Kleinberg^[2]提出了一种基于社交网络的链路预测模型，其主要思想是，通过分析网络中近邻节点的情况分析其成员可能发生哪些新的互动。由于在现实中，很多领域中的对象都是相互关联的，Taskar^[3]等人利用马尔科夫网络框架在关系型数据集上提高了准确率。Hasan^[4]等人则将监督学习的方法应用到链路预测中，并且比较了不同类别的监督学习算法的各种性能指标，结果表明大部分常用的分类算法（决策树、KNN、多层感知机、SVM 等）都有较好的性能，其中 SVM 的效果较为突出。Menon^[5]等人提出了使用矩阵分解的方法来解决图中的链路预测问题，直接从图的拓扑结构中学习潜在特征能够比无监督的方式获得更好的性能。在个性化推荐系统中，协同过滤技术的应用十分广泛，其思想是，根据评分相似邻居的兴趣和偏好产生推荐。但是，其缺点也是显而易见的，其推荐质量受到数据稀疏性的限制，其精确度和可拓展性不能得到保障。比较流行的是基于项（item）的协同过滤，Sarwar^[6]等人通过计算项之间的相似性而非用户之间的相似性大大提高了协同过滤方法的拓展性和效率。Linden^[7]等比较了基于用户的协同过滤算法以及基于聚类和搜索的推荐算法，表明还是 item-to-item 的协同过滤算法效果比较好。还有些工作利用诸如标签^[8]、地理位置^[9]等信息来研究用户兴趣的相似度，Tu^[10]等人利用新浪微博用户的标签来挖掘用户兴趣，提出了一种基于主题的标签传播模型。Qian^[11]等人将个人兴趣、人际兴趣和人际影响力三个因素结合，提出了一种基于概率矩阵分解的个性化推荐系统。

对于直接从文本中抽取用户关系的研究工作相对较少，Elson^[12]根据小说中人物的对话来分析人物关系。Qiu^[13]利用论坛文本进行情感分析来发现其中隐含的用户关系。Shen^[14]根据微博文章的内容比较了基于余弦相似度和主题的判断潜在好友的准确度。Liu^[15]通过对微博文章的关键字的提取以挖掘用户的兴趣。Xu^[16]提出了一种无监督的开放关系抽取方法用以从微博中提取关键词，闫^[17]通过在新闻标题中抽取人物实体从而建立人物关系图。赵^[18]根据用户评论采用矩阵分解的方法探索隐式社交关系，从而改进了推荐系统的性能。王^[19]利用文本信息透露的隐含社交信息结合显示社交信息，提出了一种提出了一种隐含因子图模型进而有效地进行用户群组识别。还有一些其他研究则是将用户的文本信息与其他信息结合进而分析用户兴趣和用户关系，比如利用文本信息和用户信息分析用户兴趣从而进行用户关系预测^[20]，利用用户社交数据、地理位置数据、以及文本评论这三者结合进行兴趣推荐^[21]，利用微博短文本进行聚类来对用户兴趣建模^[22]等等。

本文着重讨论了仅通过用户文本的情况下，利用深度学习的方法只根据文本特征进行好友关系预测的效果，本文不仅仅比较了与 SVM 的实验效果，同时分析了模型各个组件以及注意力机制对于预测准确率的影响。

3 数据收集

本文使用了 yelp¹商户点评网站的开放数据集，该数据集包含大量真实用户的评论等信息。我们抽取了评论数大于 50 条，好友数最多的前 5000 个活跃用户以及用户评论作为研究对象。在这 5000 个用户中，我们随机抽取 10000 对的好友以及 10000 对非好友。出于性能的考虑，对于每对好友，我们只选取他们最近发表的 5 条评论文本，合并成 1 条用户评论。我们在好友对和非好友对中各取 9000 对的评论作为训练集，其他用户对作为测试对象，即

¹ <https://www.yelp.com/dataset/>

训练集 18000 条，测试集 2000 条。表格 1 是数据集相关的信息。由表可见，每对评论的总长度是非常长的，下文的实验中，如不做特别说明，每对评论的长度限定在 500 词，即双方评论各占 250 词。模型所用的总词数限定为训练集中出现频率最高的 10000 词。

表 1 数据集相关信息

数据信息	统计
个数	各用户最近 5 条
训练集	18000 条 (9000 对)
测试集	2000 条 (1000 对)
词汇表大小	10000
每对用户评论最大长度	6921
每条用户评论最大长度	3880
样例	For the price, it's okay. I haven't stayed anywhere...

4 分析方法

4.1 模型框架

我们的目的是仅仅通过用户之间的文本信息来分析用户双方是否具备一定的好友关系。模型的输入即评论的词特征，为了避免丢失两个用户的信息，我们将好友双方的评论在输入模型之前进行拼接。图 1 描述了我们的模型的基本结构。 w_1, w_2 表示用户 A 的评论， w'_1, w'_2 表示用户 B 的评论，我们将两者的评论进行拼接后输入词嵌入层。在下文的实验中，我们使用了预训练的词向量进行初始化。因为序列长度较长，我们对词嵌入层的表示加上了一层卷积层，这使得在不影响准确率的情况下，加快了模型的训练速度。卷积后的输出再经过最重要的 LSTM 层，此时的序列仍然保持着序列特征，LSTM 能够学习到双方评论的相似性。Bahdanau^[23]等人在神经机器翻译工作中提出了注意力 (attention) 机制，该机制的引入能够让模型更加关注 LSTM 输出中两者评论中更相似的部分。模型的最后是全连接层，模型的输出为两类即是好友或非好友。

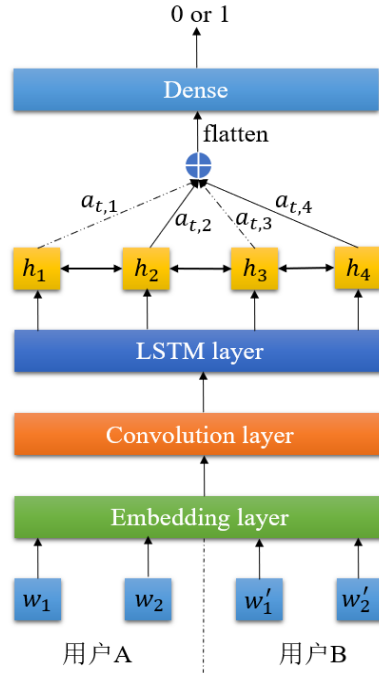


图 1 模型结构

4.2 用户的表示

为了构建基于神经网络的每个用户的表示，我们从训练集中选取出现频率最高的 10000 个词作为特征并建立词典，选取两人最近的 5 条评论，选取词典中的词作为两人的特征。比如所建文本词典为 $\text{vocab} = \{\text{'Cappuccino': 0, 'Triamisu': 1, 'Pudding': 2}\}$ ，词汇表大小就是 3。表 2 给出了用户 A 和用户 B 的文本特征。对于 SVM 来说，向量长度为词汇表大小，词的存在与否作为特征。LSTM 的输入则为两段评论的拼接。

表 2 文本特征选取

用户	用户评论	各自特征	共同特征	
			SVM 表示	LSTM 输入
用户 A	I love Pudding!	[2]	[1, 1, 1]	[2, 0, 1]
用户 B	Tiramisu and cappuccino is a perfect match!	[0,1]	[1, 1, 1]	[2, 0, 1]

从这个简单的示例中，可以看出用户 A 和用户 B 都喜欢吃西式甜点，因此我们选取字典中存在的这些甜点的名称作为他们各自的特征以及共同的特征。直觉上来说，如果两者评论中多多少少出现了相同的事物，则我们有理由认为该两人可能是潜在的好友或是拥有共同的兴趣。本文将人们最经常讨论的各个方面的关键词作为词特征并建立词典，所训练和测试的评论选取了该用户最近发表的五条长评论。

4.3 基于注意力机制和 LSTM 模型的用户关系抽取

我们首先将好友双方拼接后输入词嵌入层，然后再对词嵌入层的输出进行卷积，卷积后的结果进行了最大池化，具体公式可以简单描述为：

$$\begin{aligned}
 x_i &= W_v \cdot v_i \\
 c_i &= f(W_c x_{i:i+k_1} + b) \\
 \hat{c}_j &= \max\{c_i, \dots, c_{i+k_2}\}
 \end{aligned}$$

其中， W_v 为词嵌入矩阵， v_i 为 one-hot 形式的向量， x_i 为词嵌入层的输出， f 表示对 k_1 个词向

量进行卷积， \hat{c}_j 是池化层的输出，也是 LSTM 层的输入 x_i 。

基本的 RNN 模型的隐藏层只有一个状态 h ，而 LSTM 模型又增加了一个状态 c ，使其保存更长远的信息。LSTM 模型的输入有三个：当前时刻的输入值 x_t 、上一时刻 LSTM 的输出值 h_{t-1} 、以及上一时刻的单元状态 c_{t-1} ，LSTM 的输出有两个：当前 t 时刻的输出值 h_t 和当前时刻的单元状态 c_t 。该模型的内部有三个控制开关，一个称为遗忘门 f_t ，它决定保留多少上一时刻的信息到当前时刻；一个称为输入门 i_t ，它决定保留多少当前输入 x_t 到 c_t ；最后一个称为输出门 o_t ，它决定保留多少单元状态 c_t 到输出 h_t 。其内部运算关系可用数学符号表示为：

$$\begin{aligned}f_t &= \text{sigmoid}(W_f \cdot [h_{t-1}, x_t] + b_f) \\i_t &= \text{sigmoid}(W_i \cdot [h_{t-1}, x_t] + b_i) \\ \tilde{c}_t &= \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \\c_t &= f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \\o_t &= \text{sigmoid}(W_o \cdot [h_{t-1}, x_t] + b_o) \\h_t &= o_t \odot \tanh(c_t)\end{aligned}$$

其中， \tilde{c}_t 表示当前输入的单元状态； W_f 、 W_i 、 W_c 、 W_o 表示相应的权重矩阵， b_f 、 b_i 、 b_c 、 b_o 表示相应的偏置，它们都是该模型要训练的参数； \odot 表示点乘操作。

经过 LSTM 层后，此时模型的输出为 LSTM 每个时间步隐藏状态，我们此时将注意力机制应用于 LSTM 输出的每一个时间步，注意力机制可以捕获序列之间的依赖关系，加权求和后我们得到输出序列 i 对应的上下文向量表示 c_i ，具体公式描述如下：

$$\begin{aligned}c_i &= \sum_{j=1}^T a_{ij} h_j \\a_{ij} &= \frac{\exp(h_i^T W h_j)}{\sum_{k=1}^T \exp(h_i^T W h_k)}\end{aligned}$$

其中， h_i 为 LSTM 第 i 个时间步的输出， a_{ij} 表示第 i 个时间步与第 j 个时间步的输出进行 softmax 后的权重，这里的相似度计算函数采用的是矩阵变换，用 W 表示。最后我们还要经过一层全连接层得到最终的输出概率。

5 实验

5.1 实验设置

实验训练的数据为好友与非好友双方的评论各 9000 对，剩余 1000 对作为测试数据。每个人的评论限定为最近发表的 5 条评论，评论长度限定为 250 词。但对于 SVM 模型，我们在下文额外地列出了使用全部词汇作为特征的效果。

本文特征选择评论里出现频率最高的 10000 个词，SVM 采用词袋模型，对于 LSTM 模型，我们考虑了两种拼接方式，一种是在模型的输入时将双方的评论进行拼接，即上文所描述的框架，一种是将双方的评论分别构建两个 LSTM 层，最后将这两个 LSTM 层的输出进行拼接。注意力机制都是应用于拼接后的结果。此外，词嵌入层我们使用了预训练的词向量来进行初始化。

损失函数我们采用对数损失函数，梯度优化采用 Adam^[24]优化器。隐含变量(hidden_size)和词嵌入(embedding_size)设置的大小相同，卷积层和全连接层使用的是 relu 和 tanh 激活函数，最后的输出层使用的是 sigmoid 激活函数。LSTM 层和全连接层都使用了 dropout。超参数的值如表 3 所示。

表 3 模型的超参数设定

参数描述	设置
max length	250
embedding size	200
hidden size	200
batch size	500
learning rate	0.001
dropout	0.2
kernel size	5
filters	64
pool size	4

5.2 与基准系统结果比较

为了比较我们提出的基于注意力机制 LSTM 的模型的有效性，我们与如下基准系统进行了比较：

- **SVM**: 使用好友双方全部词的 TF-IDF(term frequency-inverse document frequency) 构建特征，并使用 SVM 模型进行用户关系预测。
- **Conv-LSTM**: 表示对词嵌入层的输出进行卷积后再输入 LSTM 层，并使用该 LSTM 模型进行预测。
- **LSTM-Concat-Att**: 是我们提出的基于注意力机制的用户关系预测模型。即在模型的输入时将双方的评论进行拼接，并将注意力机制应用于 LSTM 的输出。

表 4 不同模型准确率比较

模型	准确率
SVM	0.7500
ConvLSTM	0.7025
LSTM-Concat-Att	0.7705

对于 SVM 模型，因为输入序列的长度对于 SVM 没有影响，所以这里 SVM 使用了所有的词作为特征，而其他模型都是只取双方前 250 词拼接后作为输入。实验结果如表 4 所示。从表中可以看出，使用所有词的 TF-IDF 特征的 SVM 的效果要比进行卷积后单独使用 LSTM 的效果要好不少，但我们提出的模型要比单独使用 LSTM 的效果高了近 7 个百分点，同时也超过了 SVM 的效果。这从侧面反映了我们模型中起到关键性作用的是加入的注意力机制。

5.3 不同因素对实验结果的影响

为了充分比较模型的差异性，我们还需要分析不同的因素对于实验结果的影响，具体结果如表 5 所示。其中，

- **LSTM-Multi**: 是将双方的评论分别构建两个 LSTM 层，最后将这两个 LSTM 层的输出进行拼接，将拼接后的结果输入全连接层进行预测。
- **LSTM-Concat**: 表示输入时将双方的评论进行拼接，然后输入 LSTM 层进行预测。
- **LSTM-Multi-Att**: 表示依旧将双方评论构建两个 LSTM，不同的是，我们会将注意力机制作用于拼接后的 LSTM 输出。

- **Att**: 表示对词嵌入层的输出直接使用注意力机制并进行预测。
- **LSTM-Att**: 是指去除卷积操作, 对 LSTM 的输出运用注意力机制并进行预测。

表 5 不同因素实验结果汇总

模型	准确率
SVM	0.7500
ConvLSTM	0.7025
LSTM-Multi	0.7040
LSTM-Concat	0.6915
LSTM-Multi-Att	0.7125
Att	0.7515
LSTM-Att	0.7565
LSTM-Concat-Att	0.7705

从表中可以看出, 不管是在输入时将双方信息进行拼接还是将双方评论构建两个 LSTM 再进行拼接, 最终的效果差别不大, 但是加入注意力机制后明显是输入时拼接效果要好。虽然, 直接将注意力机制加到词嵌入层的输出的方式简单有效, 效果要比单独用 LSTM 要好不少, 但是我们的模型的效果是要超过直接使用注意力机制的效果的。最终, 我们的模型的准确率达到 0.7705, 相对于 SVM 的 0.75, 还是取得了一定的改善。值得说明的是, 我们模型的输入取的是前 250 词, 而我们模型的效果却比 SVM 使用所有词作为特征的效果要好, 这更能说明模型的优越性, 其可改进的空间也是巨大的。

通过两种模型的比较, 基本可以说明, 单用文本特征来判断是否是好友是可以取得一定的效果的, SVM 在分类任务上仍然可以不错的成绩。对于数据量大的情况, 我们的模型能仍然够充分利用有限的好友双方的文本信息, 达到明显超过 SVM 的效果。当然, 本文讨论的重点放在了文本特征上, 而其它外部特征对于判断好友关系甚至更直接, 比如两个人的共同好友数就是一个很强的特征, 拥有共同好友数较多的两人极其可能是好友, 我们甚至可以将用户的其他特征和文本特征相结合来极大地提高预测准确率。

虽然两人当前是否是好友甚至可能在数据库中已经是已知的了, 而且以上的实验是基于数据库中两人是否有确定性的好友关系的前提下完成的, 而现实中, 两人是否拥有好友关系, 并不是确定不变的, 甚至是比较主观随意的。但是, 在实际的任务中我们也并不一定需要十分确定的关系, 这种关系同时也是说不清的, 模型最终的输出可以转化为概率值, 即我们可以分析任意两对用户的文本得到两人的某些相似性。本文的实验结果依旧能够说明, 仅仅通过文本本身, 以及通过上文介绍的模型来判定两人是否拥有潜在的好友关系是有一定的效果的, 而且是有意义的, 因为当原始信息不能断定是好友, 而且其他外部特征缺乏的情况下, 我们仅仅通过文本特征就可以计算相似度以判断两人是否是潜在的好友或有共同的兴趣爱好, 这为一些推荐系统提供了一定的参考。

6 总结和展望

为了能够充分捕捉好友评论之间更深层次的信息, 本文将好友以及非好友分开进行处理, 将 LSTM 模型和注意力机制相结合, 仅仅通过分析两者的文本特征来进行好友关系预测。实验表明, 好友之间的评论确实能反映两者之间的用户关系, 注意力机制在本文任务中起到

了不可替代的作用，本文提出的好友预测模型较之于传统的 SVM 模型在准确率上有了明显的提升。

本文的主要目的是通过分析文本发现可能蕴含的好友关系，在未来的工作中，我们还会结合更复杂的注意力机制来提升准确率，同时我们还会在语义和语篇分析角度探索优化的方案，并且我们还会继续研究如何通过文本信息分析更为复杂的人物关系。

参考文献

- [1] Ramesh R. Sarukkai. Link prediction and path analysis using Markov chains 1[J]. *Computer Networks*, 2000, 33(1-6):377-386.
- [2] Liben-Nowell D, Kleinberg J. The link prediction problem for social networks[C]// Wiley Subscription Services, Inc. A Wiley Company, 2003:556-559.
- [3] Taskar B, Wong M F, Abbeel P, et al. Link prediction in relational data[C]// International Conference on Neural Information Processing Systems. MIT Press, 2003:659-666.
- [4] Hasan, Mohammad A. Link Prediction using Supervised Learning[J]. *Proc of Sdm Workshop on Link Analysis Counterterrorism & Security*, 2006, 30(9):798-805.
- [5] Menon A K, Elkan C. Link prediction via matrix factorization[C]// European Conference on Machine Learning and Knowledge Discovery in Databases. Springer-Verlag, 2011:437-452.
- [6] Sarwar B, Karypis G, Konstan J, et al. Item-based collaborative filtering recommendation algorithms[C]// International World Wide Web Conference. 2001:285-295.
- [7] Linden G, Smith B, York J. Amazon.com Recommendations: Item-to-Item Collaborative Filtering[J]. *IEEE Internet Computing*, 2003, 7(1):76-80.
- [8] Liu Z, Shi C, Sun M. FolkDiffusion: A Graph-Based Tag Suggestion Method for Folksonomies[C]// Information Retrieval Technology -, Asia Information Retrieval Societies Conference, Airts 2010, Taipei, Taiwan, December 1-3, 2010. Proceedings. DBLP, 2010:231-240.
- [9] Marlow C, Marlow C, Marlow C. Find me if you can: improving geographical prediction with social and spatial proximity[C]// International Conference on World Wide Web. ACM, 2010:61-70.
- [10] Tu H, Wang X. Mining users' interest graph in social networks with topic based tag propagation[C]// International Conference on Smart and Sustainable City. IET, 2014:282-285.
- [11] Qian X, Feng H, Zhao G, et al. Personalized Recommendation Combining User Interest and Social Circle[J]. *IEEE Transactions on Knowledge & Data Engineering*, 2014, 26(7):1763-1777.
- [12] Elson D K, Dames N, Mckeown K R. Extracting social networks from literary fiction[C]// Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2010:138-147.
- [13] Qiu M, Liu Y, Jiang J. Mining User Relations from Online Discussions using Sentiment Analysis and Probabilistic Matrix Factorization[C]// NAACL. 2011.
- [14] Shen D, Sun J T, Yang Q, et al. Latent Friend Mining from Blog Data[C]// International Conference on Data Mining. IEEE Computer Society, 2006:552-561.
- [15] Liu Z, Chen X, Sun M. Mining the interests of Chinese microbloggers via keyword extraction[J]. *Frontiers of Computer Science*, 2012, 6(1):76-87.
- [16] Xu J, Gan L, Yan Z, et al. Open Relation Extraction from Chinese Microblog Text[C]// IEEE International Conference on Data Science in Cyberspace. IEEE, 2017:673-677.
- [17] 闫咏, 赵佳鹏, 李全刚, 等. 面向文本标题的人物关系抽取[J]. *计算机应用*, 2016, 36(3):726-730.
- [18] 赵亚辉, 刘瑞. 基于评论的隐式社交关系在推荐系统中的应用[J]. *计算机应用研究*, 2016, 33(6):1628-1632.
- [19] 王中卿, 李寿山, 周国栋. 基于文本与社交信息的用户群组识别[J]. *软件学报*, 2017, 28(9):2468-2480.

-
- [20] 黄宏程, 陆卫金, 胡敏, 等. 用户兴趣相似性度量的关系预测算法[J]. 计算机科学与探索, 2017, 11(7):1068-1079.
- [21] 王啸岩, 袁景凌, 秦凤. 位置社交网络中基于评论文本的兴趣点推荐[J]. 计算机科学, 2017(12).
- [22] 邱云飞, 王琳颖, 邵良杉, 等. 基于微博短文本的用户兴趣建模方法[J]. 计算机工程, 2014, 40(2):275-279.
- [23] Bahdanau D, Cho K, Bengio Y. Neural Machine Translation by Jointly Learning to Align and Translate[J]. Computer Science, 2014.
- [24] Kingma D P, Ba J. Adam: A Method for Stochastic Optimization[J]. Computer Science, 2014.

作者联系方式: 赵赟 地址 苏州大学计算机科学与技术学院 邮编 215006 电话
15850222660 电子邮箱 yzhao666@stu.suda.edu.cn