

深 圳 大 学

硕士研究生学位论文  
开题报告书

年级 2018 级 学制 三年

姓名 张晓洁 学号 1800271040

学院（部） 计算机与软件学院

专业名称 计算机科学与技术

专业代码 081200

指导教师 廖好

研究方向 网络科学

2020 年 06 月 20 日

硕士研究生学位论文开题报告

## 撰写要求

硕士研究生中期考核通过者，进行开题报告并进入学位论文工作阶段。硕士学位论文开题报告撰写要求：

### 一、开题报告内容

1. 选题的目的和来源，课题研究的意义、学术和应用价值以及国内外研究动态；
2. 选题的基本内容、构思、创新点及初步见解；
3. 课题拟采用的研究方法和手段；
4. 课题研究程序、实验方案和预期达到的目标；
5. 论文写作进度安排及所需提供的条件、设备和经费来源。

二、开题报告会由导师指导小组（3-5人）进行集体评议，写出评语，评定考核成绩。

三、开题报告后，研究生应根据评审小组的意见，对选题方案进行修正、补充和提高，并填写《硕士研究生学位论文开题报告书》，按规定的程序报批备案后，方可进入论文写作阶段。

四、开题报告后，若学位论文课题有重大变动，应重新作开题报告，并按程序重新报批。

五、《硕士研究生学位论文开题报告书》填写内容必须属实，字迹端正清楚，由学院（部）存档备查。

# 硕士研究生学位论文开题报告

姓名	张晓洁	专业	计算机科学与技术
研究方向	网络科学		
论文题目	基于用户点评信息的推荐和餐厅倒闭可解释预测算法的研究		
开始日期	2020.06.10	完成日期	2020.06.20

## 选题报告

一、选题的来源、研究的目的意义（包括在我国应用的前景）、学术和应用价值、创新点以及国内外研究现状及水平：

### 1、研究的目的意义

随着大数据时代的来临，网络信息爆发式增长，越来越多的网络服务进入到人们的生活，用户在享受便捷网络服务的同时，会留下大量丰富的数据信息，例如用户的签到的时间、地点，用户对商品的评价，用户和商家的互动等等。这些不同类型的数据可以称为异构信息。如何充分地利用这些数据成为了网络科学一个热门的研究问题。网络科学自 2000 年之后受到了广泛关注，几乎每年在科学和自然两大国际顶级期刊都有多篇介绍相关研究进展[3-21]，并在 2009 年科学期刊专刊介绍网络科学的相关研究[12]，在现实复杂系统中，个体的功能并不能代表整个系统的功能。将复杂系统转换成网络的形式，进行分析其拓扑结构和功能的关系，构建个体的行为与系统整体功能的关系，能够有效利用大数据分析人类行为模式和社会形成。例如，在向用户推荐商品的系统中，可以根据与用户相关的用户的喜好，来像用户推荐相应的产品；在预测用户是否会购买商品的系统中，可以根据其他用户特征信息以及他们是否购买了商品的行为，来预测当前用户的购买行为。

推荐和预测系统面临的一个国际赛事挑战是 2009 年的 Netflix 公司举办的比赛，要求参赛人员或团队根据他们提供的数据，设计最好的算法进行推荐或预测。自此之后，许多公司都举办了自己的挑战赛。主持这种挑战的最著名的平台之一是 Kaggle。Netflix 提供的数据信息包括用户给电影的评分，此外，还提供了评分的时间。这些特性已经成功地应用到各种推荐和预测系统中[1,2]。在像北上广深这样的大城市里，有成千上万的餐馆，顾客自己是不可能亲自去体验到每间店的。在数据科学发展之前，我们不得不依靠人们的口碑，或者是媒体宣传才能了解到更多的参观。而现在，各种推荐系统通过用户过去的选择来预测用户未来的行为，如浏览、收集、浏览、消费等[22,23]，可以方便地为我们推荐出适合的商店。最近几年，对推荐和预测算法的挑战转移到如何利用异构信息上，例如地理位置，评论的文本内容等等[24,25]。

除了推荐和预测的准确性，人们对可解释性[26]越来越感兴趣，这些异构信息对于实现解释性有很大的作用。近年来，由机器学习和深度学习驱动的人工智能领域在发生着翻天覆地的变化。特别是深度学习，在各个领域都取得了骄人的成绩，比如人脸识别、语音识别、自然语言处理等领域的发展均深受

影响。而在预测分析方面，基于机器学习或深度学习的各类预测算法也在不断推动商业模式的变革。比如电商行业，基于大数据，根据客户点击与购买记录，利用算法推测客户喜好，展开精准营销。然而很多机器学习模型（深度学习首当其冲）的可解释性不强，这也导致在真正的商业应用中无法被广泛地采纳。这是因为企业决策者在做经营决策时无法接受一个不可解释的结论，更无法接受如果预测出来的结果并不准确，用户却不知道如何优化当前的模型。由于业务场景千变万化，没有一套通用的预测算法可以解决所有问题。既然场景、模型都有那么多的选择，企业管理者对模型的信任都会比较谨慎。因此无论我们提供给客户的解决方案的最终目标是什么，客户都需要一个可解释、可关联、可理解的解决方案，这是建立信任的必要因素，因为它代表安全、责任与可靠！此外，借助模型的可解释性，用户可以通过调整可控因素，获得最优预测结果，为企业管理者提供更多可操作的决策方法。而作为解决的提供商，我们也可以在模型的可解释性中受益，从而验证并持续改进我们的工作。

由此可以看出，将异构信息应用到网络科学服务中，无论是对于消费者还是商家，都能够提供更加准确和贴心的服务异构信息为研究人员提供了探究问题的崭新思路方法，越来越多来自不同学科的研究人员参与到信息挖掘的工作中来，将产生巨大的实际应用效果，并有助于信息科学和其他学科的深入交叉。

## 2、国内外研究现状

### (1) 推荐算法

协同过滤算法是推荐系统应用最广泛的算法，主要分为基于近邻的协同过滤算法 (memory-based CF)和基于模型的协同过滤算法( model-based CF)。基于内存的协同过滤算法又分为基于用户的协同过滤算法( UserCF) 和基于物品的协同过滤算法( ItemCF)[27]。基于用户的协同过滤算法是最早出现的推荐算法，算法首先计算和目标用户兴趣相似的用户集合，然后为目标用户推荐该相似用户集合 中用户喜欢且未接触过的物品。基于物品的协同过滤算法是目前业界应用最多的算法，该算法给用户推荐那些和他们之前喜欢的物品相似的物品。基于模型的协同过滤算法主要通过机器学习和数据挖掘模型，利用分类、回归、矩阵分解等算法提取用户和物品的隐含模式进行推荐。其中代表性的有基于贝叶斯信念网络的算法[28]、基于聚类模型的算法[29]、基于回归模型的算法[30]、基于矩阵分解模型的算法[31]等。

推荐系统是人工智能和机器学习研究的一个分支。由于深度学习在许多领域取得了巨大成功，许多研究人员开始将深度学习与推荐算法结合，解决推荐系统的各种问题( 如冷启动、稀疏性等)，提高推荐性能。当前基于深度学习的推荐算法研究分为 4 类：利用辅助信息的深度学习推荐算法；基于模型的深度学习推荐算法；动态深度学习推荐算法；基于标签的深度学习推荐算法。其中，利用辅助信息的深度学习推荐算法又可分为 3 种类型：利用辅助信息仅提取用户(或物品) 的特征表示；利用辅助信息分别提取用户和物品的特征表示；利用辅助信息提取用户和物品的共同特征表示。

20世纪 90 年代开始出现基于位置的服务和社交网络，以及基于位置的社交网络[32,33]。自这些网络出现以来，推荐系统开始使用地理信息[34,35]。最近的工作[36]采用了一组基于马尔科夫的预测器和一系列的位置推荐算法来挖掘基于位置的社交网络。另一项工作是基于位置的推荐，将协同过滤与位置[37]相结合。所得到的方法是推荐精度和计算效率之间的权衡。在[38]中，选择了一种不同的方法。将地理区域建模为两个层次系统，分别考虑了局部尺度和区域尺度。[39]研究了新增的地理信息和用户之

间的社会关系、用户之间的相似性等社会信息对新地点推荐的影响，较传统的协同过滤[40]有很大的改进。

### (2) 破产预测

破产预测是管理学和金融学文献中常见的课题。然而，现有的研究[41,42]通常局限于对流动性、偿债能力、盈利能力等财务因素的分析。此外，由于获取财务数据的挑战，所使用的数据集通常较小。使用机器学习、深度学习这种需要大量数据进行训练的模型，传统的金融数据是不适用的。随着信息技术的发展，特别是基于位置的网络服务的增长，大量的业务数据可以通过互联网收集。例如，人们可能会在他们访问的某个地点签到，在一家商店消费后，他们可以在大众点评网站上写评论，展示他们对这家商店的喜爱程度。因此，有可能利用异构信息来构建用于增强决策过程的自动业务智能工具。

### (3) 可解释预测

我们经常能在很多数据分析或 BI 产品上看到，用一些简单生硬的图表来展示预测的结果。这是一种典型的基于统计学方法的数据预测过程，通常使用线性拟合，高次曲线拟合等方法来做数据的预测。这类预测方法仅凭手头的数据，完全不需要考虑数据背后的业务逻辑。只要曲线阶次足够高，就可以做到历史数据拟合准确度无限逼近 100%。在曲线阶次限定（比如线性拟合）的情况下，这种预测由于方法简单，对预测结果还是具有一定的可解释性。但历史数据拟合的准确度不代表预测准确度，你很难说得清楚到底多少阶次的曲线拟合是更适合你的业务场景的。而且更严重的问题是，这种纯时间序列的趋势预测，只能单纯考虑时间的弱因果关系，根本没法考虑实际业务中因各种外部因素引起的数据变动，因此是一种不可增强的预测。

最近几年，深度学习和机器学习在各个领域都得到了广泛的应用，比如人脸识别、自然语言处理。我们可以发现，这些场景都是基于可伸缩、高性能的基础设施，依赖于在大量数据集上训练得到复杂的机器学习分类模型，才有可能创建和使用我们并不真正理解的决策系统。人们对于他们的信任，是基于大量的样本数据的训练和交叉检测，使得模型准确度达到可被广泛接受的程度。另一方面，在这些场景中，用户能提供的信息是全面的，无法再提供额外输入，比如，你不能让用户在进行人脸扫描的时候，再输入些其他信息来补充到算法模型里面。但以上两点，在真实的商业预测类场景里面却是不成立的。因为第一，企业可以积累一定的历史数据来供预测算法使用，但绝非能够达到人脸识别、自然语言处理模型训练这样体量的样本数据量。第二，企业业务数据的起伏波动往往是由各种外部因素共同作用引起的。我们在做预测分析的时候，不能仅仅局限于时序数据本身，而更应该深入分析业务场景，将各种具有因果关系的外部因素量化后加入到预测模型中来，这样的预测才是真正跟业务接轨的。而这种迭代优化的能力是人脸识别等场景所不具备的。第三，企业数据在收集过程中，可能存在一些系统性的偏差，这可能会导致在预测、训练过程中找到一些虚假关联，做出错误决策，因此在做预测分析时，往往需要对结果的信任和接受作出解释。

因此在与企业相关的预测中，解释性是非常重要的。借助模型的可解释性，用户可以通过调整可控因素，获得最优预测结果，为企业管理者提供更多可操作的决策方法。而作为解决的提供商，我们也可以在模型的可解释性中受益，从而验证并持续改进我们的工作。

## 3、创新点分析

- (1) 在推荐算法中，我们在不获取用户当前位置的情况下，使用 DBSCAN 算法预测用户的位置。既保护了用户的隐私，又提高了推荐的准确性。
- (2) 在可解释预测模型中，相比传统的破产预测使用的金融数据，我们使用的数据来自网络服务，数据规模大且更容易获取。
- (3) 在可解释预测模型中，从评论文本数据中构造中文概念集，提高了预测的准确性和解释性。
- (4) 在可解释预测模型中，将预测任务和解释任务进行联合学习，提高了模型的泛化能力。

二、论文研究的主要内容，方案和拟采用的研究方法、手段；已进行的科研工作基础和已具备的科学实验条件（包括文献资料及主要实验仪器设备准备情况等），对其他单位的协作要求；论文总工作量（估计），论文初稿的进度以及预期结果：

### 1、论文研究的主要内容，方案和研究方法

#### 1.1 基于扩散的位置感知推荐系统

##### 1.1.1 二分网络

大多数数据可以用网络(或称为图)表示。网络由节点组成，在[43]中，当两个节点之间有关系时，通过一条链路连接起来。在这项工作中，我们使用具有两种类型节点的数据：用户和项目。链接只能发生在用户和项目之间(即不能发生在两个用户或两个项目之间)。这就是我们定义的二分图网络。为了简单起见，我们对非用户节点都统称位项目，它们可以是与用户交互的任何东西。例如，用户享受的服务、购买的产品、发布的评论信息等。

我们用拉丁字母表示用户，用希腊字母表示项目。与数据相关的时间用  $t$  表示。用户集被定义为  $U$ ，项目集用  $I$  表示。网络的邻接矩阵  $A$  的元素表示用户和项目之间的关系。如果用户  $i$  连接到项目  $\alpha$ ，则有  $a_{i\alpha}=1$ ，否则则有  $a_{i\alpha}=0$ 。用户  $i$  的用户的度  $k_i(t)$  表示当时间为  $t$  时，连接到该用户的项目数。相似地，项目的度  $k_\alpha(t)$  代表  $t$  时刻连接到项目上的用户数量。

##### 1.1.2 扩散推荐算法

在第一部分研究内容中，我们是以基于概率扩散[44]的推荐系统为基础进行算法改进的。选择了两种阔算推荐算法，第一个是基于随机游走过程的物质扩散推荐算法，首先每一个商品把自己的能量平均分给所有购买过它的用户。用户的能量值则是从所有商品所得到的能量值得总和；接下来，每一个用户再把自己的能量平局分给所有购买过的商品，商品的能量则是从所有用户收到的能量值得总和。它倾向于推荐那些度较大（较流行）的物品，相当于一个凸透镜，将用户的视野汇聚在那些较流行的节点上，具有较高的推荐精度。而基于热扩散过程 [45]的方法首先每一个用户的温度等于所有他购买过的商品的温度的平均值，接下来每一个商品的温度等于所有购买过他的用户的温度的平均值。热传导倾向于推荐那些度较小（较不流行）的节点，相当于一个凹透镜，把用户的视野发散到了那些较不流行的物品，但更倾向于推荐对象的多样性。由于两者都是基于扩散的物理过程，所以它们可以优雅地合并在一起。这两个过程的合并产生了一种同时比两个过程分开更精确和更多样化的方法。

THybridS 的推荐分数首先通过传播和过程获得，然后根据最近的网络活动调整分数。在数学上，传播矩阵为：

$$W_{\beta}(t) = \frac{1}{k_{\alpha}(t)^{1-\lambda} k_{\beta}(t)^{\lambda}} \sum_{j=1}^U \frac{a_{j\alpha}(t) a_{j\beta}(t)}{k_{\alpha}(t)}$$

项目  $\alpha$  对用户  $i$  的推荐分数为:

$$r_{\alpha}^{(i)}(t) = \sum_{\beta=1}^I W_{\beta}(t) a_{i\beta}(t)$$

## 1.2 基于概念的餐厅倒闭可解释预测

### 1.2.1 注意力协同选择器

注意协同选择器是可以从众多数据中选择出重要信息的模型。在我们的任务中，分别使用了两个类型的选择器：评论层选择器和概念层选择器。其中，评论层选择器负责从多条评论中选择出最重要的一条评论；概念层选择器负责从一条评论中选择出重要概念。

#### (1) 评论层选择器

评论权重矩阵：对于给定的一个用户评论向量  $r_{u,i}$  ( $r_{u,i} \in R^{l_r \times l_q}$ ) 和一个餐厅评论向量  $r_{v,j}$  ( $r_{v,j} \in R^{l_r \times l_q}$ )，评论权重矩阵计算如下：

$$AR_{i,j} = F(r_{u,i})^T M_r F(r_{v,j})$$

其中  $M_r \in R^{l_q \times l_q}$ ,  $AR \in R^{l_r \times l_r}$ ,  $F$  是一个  $l$  层的前馈神经网络。

池化操作：使用池化操作来选择  $AR$  矩阵每一行每一列的最大值，分别用来表示  $r_{u,i}$  和  $r_{v,j}$ 。

$$u_i = \left( G \left( \max_{col} (AR) \right) \right)^T r_{u,i}$$

$$v_j = \left( G \left( \max_{row} (AR) \right) \right)^T r_{v,j}$$

其中  $G(\cdot)$  是将向量  $u = (u_1, \dots, u_{l_r})$  和  $v = (v_1, \dots, v_{l_r})$  转化为离散分布的函数。

**Straight-Through Gumbel-Softmax.:** 通常， $G(\cdot)$  使用 softmax 函数。由于我们想进一步操作选定的评论，softmax 返回一个不可微的向量。因此，我们使用 Straight-Through Gumbel-Softmax 来解决这个问题。

$$b_i = \frac{\exp(u_i + g_i)}{\sum_{j=1}^{l_r} \exp(\frac{u_j + g_j}{\tau})}$$

其中， $\tau$  是一个参数，当它为 0 时， $b_i$  是一个独热向量。 $g_i$  是一个 Gumbel noise。在前馈过程中， $b_i$  会被转化为一个独热向量  $x_i$ 。

$$x_i = \begin{cases} 1, & i = \text{argmax}_j(b_j) \\ 0, & \text{otherwise} \end{cases}$$

在向后传递过程中，Straight-Through Gumbel-softmax 的梯度是连续的，所以我们继续使用 Gumbel(.) 来计算向量。

$$\dot{r_u} = (\text{Gumbel})^T R_u$$

$$\dot{r_v} = (\text{Gumbel})^T R_v$$

其中， $\dot{r_u}$  和  $\dot{r_v}$  分别是用户和餐厅在评论层的向量表示。

#### (2) 概念层选择器

为了在用户和餐厅之间进行更细粒度的比较和提供更多信息的交互，我们对选定的用户评论  $r_{u,i}$  和餐厅评论  $r_{v,j}$  进行概念级建模。 $c_{u,i}$  表示  $r_{u,i}$  第  $i$  个概念， $c_{v,j}$  表示  $r_{v,j}$  第  $j$  个概念， $l_c$  是每条

评论文本中包含的概念的最大权重。概念权重矩阵  $AC$  的计算如下：

$$AC_{ij} = F(c_{u,i})^T M_c F(c_{v,j})$$

在使用平均池化操作从  $AC$  中计算出表示用户和餐厅的表示向量  $\vec{c}_u$  和  $\vec{c}_v$

$$\begin{aligned}\vec{c}_u &= (\text{Gumbel}(\text{avg}_{\text{col}}(AC)))^T c_u \\ \vec{c}_v &= (\text{Gumbel}(\text{avg}_{\text{row}}(AC)))^T c_v\end{aligned}$$

### (3) 多指针学习

由于在评估商店时可以同时考虑评论和概念，所以我们支持聚合多个指针。以用户  $u$  为例，用户的协同注意力嵌入向量表示  $\bar{u} = [r_u : \vec{c}_u]$ ，由评论级用户嵌入和概念级用户嵌入组成。在多指针设置中，我们使用不同的 Gumbel 噪声运行选择器多次，得到多个样本： $\{\bar{u}_1, \dots, \bar{u}_p\}$ 。同样，我们获得了一组共同关注项嵌入的样本： $\{\bar{v}_1, \dots, \bar{v}_p\}$ 。然后我们使用一个非线性层  $\text{ReLU}(\cdot)$  来聚合它们。

$$\begin{aligned}x_u &= \text{ReLU}(W[\bar{u}_1, \dots, \bar{u}_p] + b) \\ x_v &= \text{ReLU}(W[\bar{v}_1, \dots, \bar{v}_p] + b)\end{aligned}$$

### 1.2.2 Factorization machine(FM)模型

FM 用于实现不同特征之间的两两交互  $[x_u, x_v]$ ，预测公式如下。

$$\hat{y} = f(x) = w_0 + \sum_{i=1}^n w_i x_i + \sum_{i=1}^n \sum_{j=i+1}^n \langle p_i, p_j \rangle x_i x_j$$

由于任务是预测餐厅关闭情况，这是一个二元预测问题，所以通常使用 sigmod 交叉熵损失函数。

$$L_y = \frac{1}{2|\Omega|} \sum_{(u,i) \in \Theta} (-[y \log \hat{y} + (1-y) \log (1-\hat{y})])$$

其中， $\Omega$  代表了训练集， $y$  是餐厅真实倒闭情况。

### 1.2.3 Gated Recurrent Unit(GRU)模型

近期研究表明，LSTM 和 GRU 在生成文方面表现良好。考虑到 GRU 在相似性能下具有较少的参数和较高的计算效率，我们的实验使用 GRU 来生成解释文本。

$$p(l_t | l_1, l_2, \dots, l_{t-1}, \hat{y}) = \zeta(h_t)$$

其中， $l_t$  是生成的在  $t$  时刻的单词。 $\zeta(\cdot)$  是一个 softmax 函数。

$$\zeta(z^{(i)}) = \frac{e^{z^{(i)}}}{\sum_{n=1}^N e^{z^{(i)}}}$$

$h_t$  是  $t$  时刻的隐藏状态，依赖于  $t-1$  时刻的隐藏状态。

$$h_t = \text{GRU}(h_{t-1}, l_t)$$

出示隐藏状态  $h_0$  表达如下：

$$h_0 = \tanh(W_u x_u + W_v x_v + W_u \hat{y} + b)$$

然后将隐藏状态输入到最终的输出层，生成时间为  $t$  的单词分布。

$$d_t = \zeta(W_d h_{t-1} + b_d)$$

解释任务的损失值由两部分组成，第一部分是生成文本整体的相似性：

$$L_o = \frac{1}{|\Omega|} \sum_{(u,v) \in \Theta} \sum_{t=1}^T (-\log d_{t,\bar{l}_t})$$

另外一个损失值是概念相似性的损失值：

$$L_c = \frac{1}{|\Omega|} \sum_{(u,v) \in \Omega} \sum_{t=1}^T (\max(-\delta_i \log d_{t,i}))$$

## 2、已进行的科研工作基础

### 2.1 基于扩散的位置感知推荐系统

#### 2.1.1 DBSCAN 算法

在大多数数据集中，项目的位置是给定的。但出于隐私考虑，用户的位置尚未明确。在这项工作中，我们假设每个用户都有一个他经常去的最喜欢的区域。我们将此区域称为用户的理想位置。根据这个假设，我们可以将用户的位置定义为其历史签到记录的平均位置。然而，这种方法有一个主要的缺点：如果用户住在纽约，并定期评论那里的商品，他的平均位置将位于纽约的某个地方。但如果他去巴黎度假1周，在那里点评了商品，他的平均位置突然间在大西洋的某个地方。

为了解决这个问题，我们设计了一个DBSCAN聚类算法，根据用户的历史签到记录，计算用户的当前的位置。算法的示意图如图1，将用户的历史签到记录看成样本点，随机选择一个点为圆心，以指定半径画圆，规定圆内包含的样本点最少个数阈值。如果画出来的圆中包含的样本数不小于阈值，则将圆心转移到圆内的其他样本点。不断地遍历、扩散，直到遍历完所有样本点，会把用户的历史签到记录划分为多个聚落区域和一些离散的点。接下来，从这些聚落区域中选择出一个最能代表用户稳定访问的区域，如图1(b)。每个区域内的项目按签到时间进行排列，箭头上标注两个项目签到的时间间隔。然后计算每个区域中时间间隔至少为M天时间间隔数。数量最大的区域被选为用户*i*的最优区域，其圆心位置表示用户的当前位置。

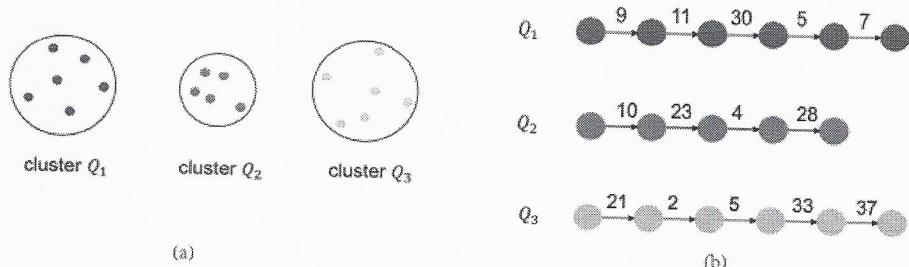


图1展示了DBSCAN聚类算法。对于给定的用户*i*，(a)显示了他的不同聚落区域。(b)显示聚落区域中以天为单位的签到时间间隔。

#### 2.1.2 Geo-ThybirdS

在用户驱动的数据集中，地理位置很重要。例如，餐馆和商店有一个物理位置。如果没有地理信息，推荐结果可能会很棘手。人们倾向于去附近的地方，并定期到某些地方去。这个信息是非常重要的，因为我们可以假设人们趋向于定期去同一个地方，因为他们住或工作的地方离那个地区很近，或者他们特别喜欢这个地区。在此基础上，我们提出了一种基于地理位置的Geo-ThybirdS算法。计算公式如下：

$$g_\alpha^{(i)} = r_\alpha^{(i)} \exp(d_{ia}\theta)$$

其中，\$d\_{ia}\$是用户*i*和项目*a*的距离，\$\theta\$是调节参数。

#### 2.1.3 实验

这部分实验使用了国外三个签到网站的数据作为数据集，分别是Yelp, Foursquare和Brightkite，数据信息包括用户id，项目id，签到时间，用户和商品的经纬度。表1是三个数据集的基本信息。

	<b>U</b>	<b>I</b>	<b>L</b>	$\langle k_i \rangle$	$\langle k_\alpha \rangle$
Yelp	123368	41958	804789	6.5	19.6
Brightkite	37303	15651	201308	92.3	3.4
Foursquare	2293	61852	211670	5.4	12.9

表 1：三个数据集的基本信息：用户数量 U，项目数量 I，链接数量 L，用户平均度  $\langle k_i \rangle$ ，项目平均度  $\langle k_\alpha \rangle$ 。

除了 THybridS 作为基线方法之外，还选择了另外 3 个对比方法：第一个是 Fusion 算法[46]，该算法使用两种不同的评分，一种基于用户相似度，另一种基于地理位置。分数分别计算，然后合并在一起。第二个是 closest-item，我们在 THybridS 中添加了空间距离，但是我们也想将空间距离本身的性能作为基准性能来考虑。我们设计了一个 Closest-item 方法，它只使用地理信息进行推荐。它根据产品与用户之间的距离对产品进行排序，然后生成推荐列表。第三个是 SVDPP，是传统的机器学习算法。

在图 2 展示了在三个数据集上，改变 lambda 参数和距离参数，recall，precision 和新颖度的热度图。可以看出，这三个指标的趋势基本一致。

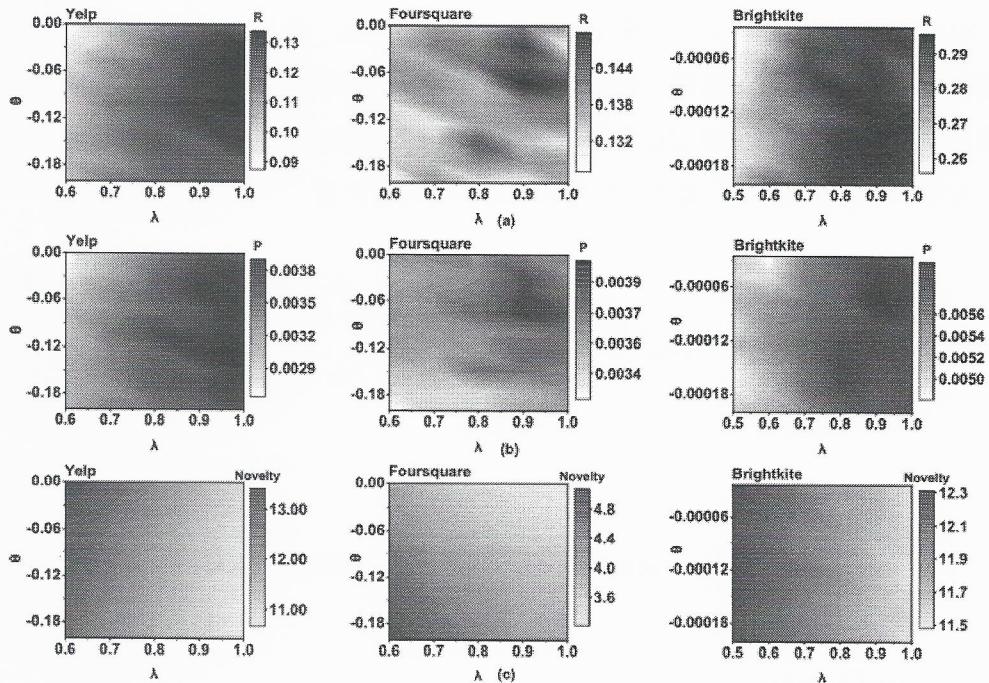


图 2：Yelp, Foursquare, Brightkite 数据集上的 precision、recall 和新颖度热度图

表 2 展示了在三个数据集，我们的方法和 THybridS 的比较结果。在这三个数据集中，recall 值有明显的提高：Yelp 的 recall 优化了为 5.5%，Foursquare 的优化了 8.7%，Brightkite 的优化为 2.5%。另外，我们的方法在 precision 上始终优于 THybrids。具体来说，在 Yelp、Foursquare 和 Brightkite 上，Geo-THybridS 分别比 THybrids 提高了 8.3%、2.6% 和 1.8%。AUC 并没有因为在这三个地方都添加了地理修改而得到改善。这是因为数据集中项目的数量很多，项目的排名不会显著影响 AUC 的值，例如，项目的排名从 20 名变化到 50 名，AUC 评分不会发生很大变化。但对于用户来说，这是非常重要的，所以我们使用 recall 作为主要的评估指标。Mean distance 指标是用来评估推荐列表上的项目距离用户的平均距离，可以明显的看出，我们的方法推荐出来的产品与用户的距离更近，这也符合我们的常识。

		Accuracy				Diversity		Mean distance
		P	R	AUC	NDGC	Novelty	Coverage	
Yelp	THybridS	0.0036	0.127	0.907	0.093	11.4	%6.8	46.39
	Geo-THybridS	<b>0.0039</b>	<b>0.134</b>	0.896	<b>0.096</b>	10.7	%6.8	<b>5.09</b>
Foursquare	THybridS	0.0038	0.138	0.888	0.107	4.1	%2.6	7.1
	Geo-THybridS	<b>0.0039</b>	<b>0.150</b>	0.881	<b>0.115</b>	4.1	%2.6	<b>4.73</b>
Brightkite	THybridS	0.0055	0.283	0.877	0.172	11.3	%1.8	757.9
	Geo-THybridS	<b>0.0056</b>	<b>0.290</b>	0.879	<b>0.179</b>	11.3	%1.8	<b>537.1</b>

表 2: THybridS 和 GeoTHybridS 的准确度度、多样性和平均距离的比较。推荐列表的长度 N = 50。

我们不仅和 THybridS 进行了对比，而且还跟另外三个基线算法进行比较。如图 3 所示，性能最好的两个算法是算法是 THybridS 和 Geo-THybridS。另外两种算法，fusion 算法和 closest-item，也使用了地理数据。结果是明确的，我们的算法在准确性方面明显优于其他方法。在 recall 指标方面，我们的算法比 THybridS 算法的表现优化了 3.3%，并且比 fusion 算法的表现提高了 3 倍，比 closest-item 算法的表现优化了 5 倍，比 SVDPP 的表现好 10 倍。其他三个精度指标 Precision, F1 和 NDGC 也通过我们的方法得到了显著的改进。

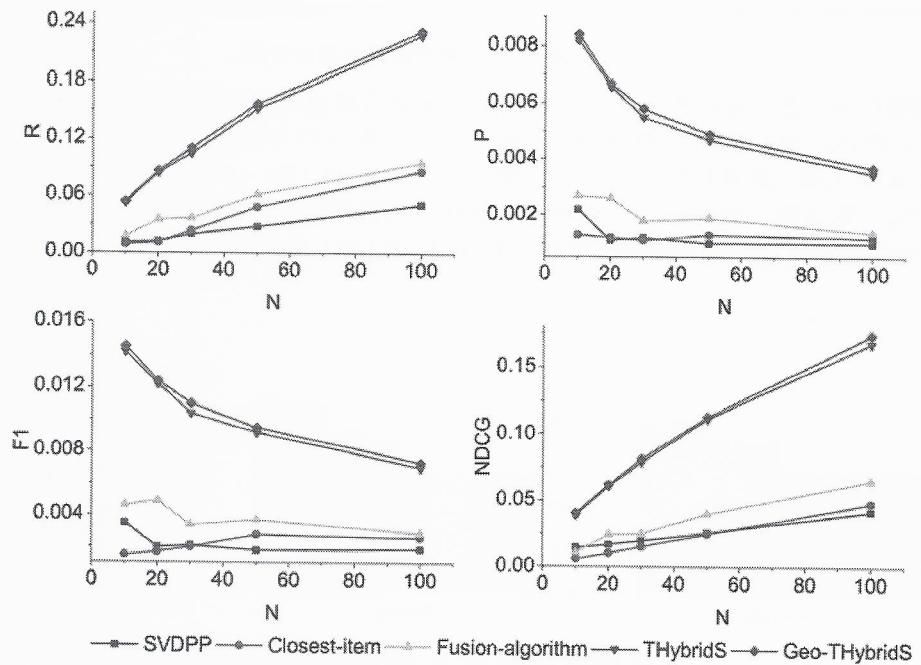


图 3: Yelp 数据集上，Geo-THybridS 和其他四个基线方法在准确度的比较。

## 2.2 基于概念的餐厅倒闭可解释预测

### 2.2.1 构造中文概念集合

我们在这部分工作中提出了中文概念，概念的定义是评论文本中的关键词，可以是名词、也可以是形容词。我们为每个数据集构造一个中文概念集合，方法如下：从数据集的评论文本中提取出所有名词和形容词，然后选择出出现频率比较高的前 K 个词，此处 K 的大小取决于数据集规模的大小。然后再去除其中和我们研究主题密切相关的无用词，比如“餐厅”、“味道”等。

### 2.2.2 多任务联合学习

我们的算法要为用户提供预测功能和解释功能，目前很多的多任务研究都是采用分开训练的方法，这样可能会导致模型过拟合。为了解决这个问题，我们将多任务进行联合训练。其中，预测任务的损失值为  $L_y$ ，解释任务的损失值由两部分，分别是  $L_o$  和  $L_c$ 。我们将这三种损失综合起来，形成我们的多任务学习模型的最终目标函数。经过实践，我们选择性能最好的 Adam 优化器。

$$L = p_1 L_y + g_1 (\lambda_o L_o + \lambda_c L_c) + \lambda_l \|\Theta\|_2^2$$

其中,  $p_1$  和  $g_1$  分别是预测任务和解释任务的权重,  $\lambda_o$  和  $\lambda_c$  是整体相似性和概念相似性的权重。最后  $\lambda_l \|\Theta\|_2^2$  是调节参数。

### 2.2.3 实验

在这部分研究内容中, 我们使用的数据集是大众点评网站 2010 年间用户对商店的评论文本。因为我们的模型需要大量的数据进行训练, 为了保证实验效果的有效性, 我们选择了数据量排名前三名的城市的数据, 分别是上海、北京、广州。这三个城市的基本数据统计如表 3。

	reviews	users	shops	restaurants	closed
上海	581713	126158	11508	10251	3312
北京	361635	75283	8476	5067	1308
广州	87166	20532	2247	1932	509

表 3: 三个城市的统计数据。

预测任务方面, 我们选择了三个常用的机器学习算法和三个深度学习算法作为基线方法, 分别是 CNN, RNN, NRT, GBDT, SVM, LR。图 3 表明, 我们的方法在三城市的 AUC 中始终优于基线方法。DCA 在上海、北京和广州分别比基准方法提高了  $18.12\% \sim 50.74\%$ 、 $19.41\% \sim 59.41\%$  和  $19.20\% \sim 49.52\%$ 。这证明了我们模型的有效性。另外, 我们还剔除了 DCA 模型中的概念层选择器, 构造了 DCA-C 变体, 实验结果标明 DCA-C 的 AUC 比 DCA 会低一些, 这也说明了概念在预测任务的重要性。

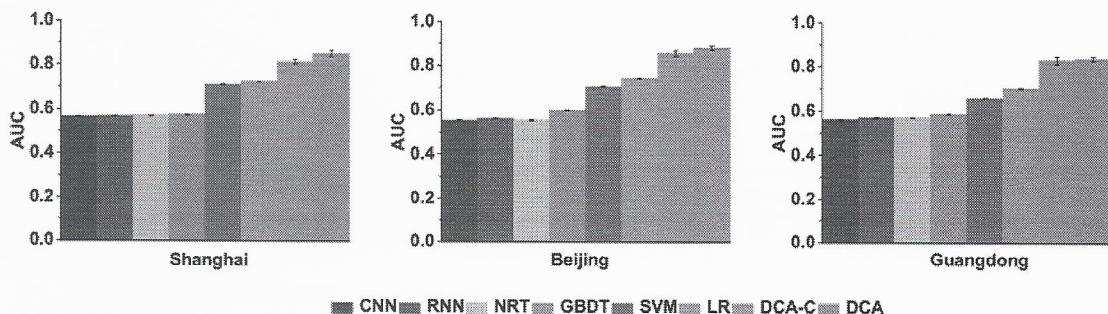


图 4: 三个数据集中, 我们的方法和基线方法在准确性上的比较结果。

解释任务方面, 我们选择了 Lexrank 和 NRT 算法作为基线算法。首先分析整理的文本相似性, 表 4 显示, 我们的方法在 BLEU 和 ROUGE 指标上始终优于基线。以 BLEU 数据为例, 我们的方法在广州数据和北京数据上分别比基线提高  $2.99\% - 36.07\%$  和  $7.23\% - 49.5\%$ , 在上海数据上分别提高  $0.67\% - 11.64\%$ 。我们还给两个基线算法添加了概念信息, 分别构造了变形 Lexrank+C 和 NRT+C, 实验结果标明, 添加了概念信息的基线方法也比原始的基线方法有提升, 这也充分说明了概念在解释任务起到的积极作用。

Method	BLEU	ROUGE_L			ROUGE_2			ROUGE_L			ROUGE_SU4		
		R	P	F1									
Shanghai													
LexRank	1.349	2.681	3.669	2.73	0.769	0.925	0.788	2.62	3.592	2.676	0.978	1.336	0.943
LexRank+C	1.433	3.423	6.512	3.244	0.921	1.282	0.904	2.998	5.056	2.968	1.057	1.564	1.098
NRT	1.496	4.076	8.356	4.859	0.959	1.778	1.049	4.064	8.31	4.839	1.083	2.399	1.143
NRT+C	1.499	4.121	8.422	4.878	1.034	1.834	1.096	4.165	8.403	4.912	1.132	2.41	1.244
DCA-C	1.487	5.811	8.659	5.844	1.226	1.921	1.247	5.677	8.234	5.175	2.036	2.987	1.549
<b>DCA</b>	<b>1.506</b>	<b>5.994</b>	<b>9.014</b>	<b>6.07</b>	<b>1.829</b>	<b>2.587</b>	<b>1.741</b>	<b>6.009</b>	<b>8.966</b>	<b>6.036</b>	<b>2.261</b>	<b>3.227</b>	<b>1.889</b>
Beijing													
LexRank	1.19	2.113	3.982	2.393	0.53	1.233	0.62	2.109	3.936	2.393	0.115	1.251	0.127
LexRank+C	1.373	2.986	4.532	3.023	0.877	1.785	1.097	2.930	5.011	3.212	0.324	1.778	0.341
NRT	1.659	3.631	7.796	4.398	0.818	1.704	0.98	3.631	7.796	4.398	0.848	2.182	0.987
NRT+C	1.685	3.878	7.931	4.657	1.324	1.734	1.045	3.895	7.899	4.563	1.098	2.233	1.151
DCA-C	1.706	4.414	7.169	4.696	1.185	1.748	1.117	4.37	7.106	4.645	1.244	2.136	1.106
<b>DCA</b>	<b>1.779</b>	<b>4.908</b>	<b>7.918</b>	<b>4.983</b>	<b>1.194</b>	<b>2.112</b>	<b>1.2</b>	<b>4.872</b>	<b>7.9</b>	<b>4.938</b>	<b>1.465</b>	<b>2.636</b>	<b>1.281</b>
Guangzhou													
LexRank	1.317	4.233	11.142	5.461	0.477	1.001	0.57	0.4233	11.142	0.546	0.657	1.523	0.782
LexRank+C	1.728	6.132	11.534	6.989	1.768	3.745	2.113	5.766	11.53	6.731	1.822	4.702	1.935
NRT	1.74	4.015	11.46	5.285	0.651	2.346	0.906	4.007	11.424	5.272	0.653	3	0.862
DCA	The environment was <b>OK</b> . The taste of the dishes was too bad to <b>flattering</b> , but when I ate the braised pork, I wanted to spit it out.												
Case 2	There was really nothing to eat. Although meat is expensive, it can't take starch to wrap everything! All the dishes were <b>disappointing</b> . The <b>attendant</b> was even more <b>outrageous</b> . She couldn't understand Mandarin. She asked for money without checking the bill. I wouldn't go again.												
Lexrank	There was really nothing to eat. The service was <b>outrageous</b> .												
NRT	There was nothing special about the taste, the service was very general, I wouldn't go there again.												
DCA	There was really nothing to eat. The dishes were <b>disappointing</b> . The attendant was <b>outrageous</b> and had a bad attitude. I wouldn't go again.												

表 4: 三个城市的解释文本评估结果。

除了评估文本整体相似性，我们还研究了生成的文本中包含的概念的相似性。表 5 展示了两个例子，case 是我们选择出来的 ground truth 解释文本，粗体字是概念。从表中可以看出，我们的方法生成概念更接近真实情况。

Case 1	The <b>environment</b> was <b>OK</b> , but the dishes were not <b>flattering</b> . It was too <b>bad</b> . I really wanted to spit it out when I ate the braised pork. It was no <b>salty</b> taste. Other dishes were not so <b>good</b> .
Lexrank	The <b>environment</b> was <b>OK</b> . The dishes was too <b>bad</b> .
NRT	The <b>environment</b> was <b>good</b> . The food was <b>terrible</b> .
DCA	The <b>environment</b> was <b>OK</b> . The taste of the dishes was too bad to <b>flattering</b> , but when I ate the braised pork, I wanted to spit it out.
Case 2	There was really nothing to eat. Although meat is expensive, it can't take starch to wrap everything! All the dishes were <b>disappointing</b> . The <b>attendant</b> was even more <b>outrageous</b> . She couldn't understand Mandarin. She asked for money without checking the bill. I wouldn't go again.
Lexrank	There was really nothing to eat. The service was <b>outrageous</b> .
NRT	There was nothing special about the taste, the service was very general, I wouldn't go there again.
DCA	There was really nothing to eat. The dishes were <b>disappointing</b> . The attendant was <b>outrageous</b> and had a bad attitude. I wouldn't go again.

表 5: Lexrank、NRT 和我们的方法生成的解释文本示例。

### 3、论文总工作量（估计），论文初稿的进度以及预期结果：

本课题研究时间为 2020 年 3 月到 2021 年 4 月。

- (1) 2020.03-2020.05: 对相关的论文进行研究，需要历时两个月；
- (2) 2020.06-2019.07: 设计异构信息的科技是预测算法；
- (3) 2020.07-2020.9: 实验对比设计的算法与最新发表的可解释预测算法的精确度和解释性能等指标；
- (4) 2020.09-2020.10: 完成论文初稿的撰写；
- (5) 2020.11-2020.12: 完成专利的撰写。

后期工作：主要工作是撰写论文（时间为 2019 年 12 月—2020 年 4 月）。具体安排如下：

- (1) 整理实验并完成论文初稿；(2) 论文修改；(3) 论文定稿。

预期结果：1、发表期刊论文 2 篇，CCF-B 类会议论文 1 篇。

2、申请发明专利 2 篇。

参考文献：

- [1] L. Lu, M. Medo, C. H. Yeung, Y.-C. Zhang, Z.-K. Zhang, T. Zhou, Recommender systems, Physics

- [2] F. Yu, A. Zeng, S. Gillard, M. Medo, Network-based recommendation algorithms: A review, *Physica A: Statistical Mechanics and its Applications* 452 (2016) 192-208.
- [3] S. P. Borgatti, A. Mehra, D. J. Brass, G. Labianca, Network Analysis in the Social Sciences, *Science* 323, 892(2009).
- [4] V. Colizza, A. Barrat, M. Barthelemy, A. Vespignani, The Modeling of Global Epidemics: Stochastic Dynamics and Predictability, *Bull. Math. Biol.* 68, 1893(2006).
- [5] C.H. Yeung, D. Saad and K. Y. M. Wong, From the physics of interacting polymers to optimizing routes on the London Underground, *Proc. Natl. Acad. Sci. USA* 110, 13717 (2013).
- [6] L. Lu, M. Medo, C. H. Yeung, Y.-C. Zhang, Z.-K. Zhang, T. Zhou, Recommender Systems, *Phys. Rep.* 519, 1 (2012).
- [7] D. J. Watts, S. H. Strogatz, Collective dynamics of “small-world” networks, *Nature* 393, 440(1998).
- [8] A.-L. Barabasi, R. Albert, Emergence of Scaling in Random Networks, *Science* 286, 509(1999).
- [9] S. Fortunato, Community detection in graph, *Phys. Rep.* 486, 75 (2010).
- [10] A. Zeng, S. Gualdi, M. Medo and Y.-C. Zhang, Trend prediction in temporal bipartite networks: the case of MovieLens, Netflix, and Digg, *Advs. Complex Syst.* 16, 1350024 (2013).
- [11] A. Zeng, S. Gualdi, M. Medo and Y.-C. Zhang, Trend prediction in temporal bipartite networks: the case of MovieLens, Netflix, and Digg, *Advs. Complex Syst.* 16, 1350024 (2013).
- [12] Q.-M. Zhang, A. Zeng and M.-S. Shang, Extracting the information backbone in online systems, *PLoS One* 8(5), e62624 (2013).
- [13] L. Lu and T. Zhou, Link Prediction in Complex Networks: A Survey, *Physica A* 390, 1150 (2011).
- [14] K.-I. Goh, A.-L. Barabási, Burstiness and memory in complex systems, *Europhys. Lett.* 81 (4) (2008) 48002.
- [15] Y. Borghol, S. Mitra, S. Ardon, N. Carlsson, D. Eager, A. Mahanti, Characterizing and modelling popularity of user-generated videos, *Perform. Eval.* 68(11) (2011) 1037–1055.
- [16] Y.-H. Eom, S. Fortunato, Characterizing and modeling citation dynamics, *PLoS One* 6 (9) (2011) e24926
- [17] S. Fortunato, A. Flammini, F. Menczer, Scale-free network growth by ranking, *Phys. Rev. Lett.* 96 (21) (2006) 218701.
- [18] D. Goldberg, D. Nichols, B. M. Oki and D. Terry, *Commun. ACM* 35 (1992) 61.
- [19] J. B. Schafer, D. Frankowski, J. Herlocker and S. Sen, *Lect. Notes Comput. Sc.* 4321 (2007) 291.
- [20] S. Maslov and Y.-C. Zhang, *Phys. Rev. Lett.* 87 (2001) 248701.
- [21] Y.-C. Zhang, M. Blattner and Y.-K. Yu, *Phys. Rev. Lett.* 99 (2007) 154301.
- [22] A. Said, A. Bellogn, J. Lin, A. D. Vries, Do recommendations matter?: news recommendation in real life, in: Companion Publication of the ACM Conference on Computer Supported Cooperative Work & Social Computing, ACM, 2014.
- [23] H. Liao, M. S. Mariani, M. Medo, Y.-C. Zhang, M.-Y. Zhou, Ranking in evolving complex networks, *Physics Reports* 689 (2017) 1-54.
- [24] Z. Lu, Z. Dou, J. Lian, X. Xie, Q. Yang, Content-based collaborative filtering for news topic recommendation, in: Twenty-ninth AAAI conference on artificial intelligence, 2015.

- [25] G. Adomavicius, A. Tuzhilin, Context-aware recommender systems, in: Recommender systems handbook, Springer, 2011, pp. 217-253.
- [26] Rago A, Cocarascu O, Toni F, et al. Argumentation-Based Recommendations: Fantastic Explanations and How to Find Them[C]. international joint conference on artificial intelligence, 2018: 1949-1955.
- [27] Tacchella A, Cristelli M, Caldarelli G, et al. A new metrics for countries' fitness and products' complexity[J]. Scientific reports, 2012, 2: 723.
- [28] Miyahara K, Pazzani M J. Collaborative filtering with the simple Bayesian classifier[C]//Pacific Rim International conference on artificial intelligence. Springer, Berlin, Heidelberg, 2000: 679-689.
- [29] Chee S H S, Han J, Wang K. Rectree: An efficient collaborative filtering method[C]//International Conference on Data Warehousing and Knowledge Discovery. Springer, Berlin, Heidelberg, 2001: 141-151.
- [30] Canny J. Collaborative filtering with privacy via factor analysis[C]//Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2002: 238-245.
- [31] Koren Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model [C] //Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2008: 426-434.
- [32] A. Colorni, M. Dorigo, V. Maniezzo, et al., Distributed optimization by ant colonies, in: Proceedings of the rst European conference on articial life, Vol. 42, Cambridge, MA, 1992, pp. 134-142.
- [33] M. Dorigo, V. Maniezzo, A. Colorni, et al., Ant system: optimization by a colony of cooperating agents, IEEE Transactions on Systems, man, and cybernetics, Part B: Cybernetics 26 (1) (1996) 29{41.
- [34] Q. Li, Z. Yu, X. Xing, Y. Chen, W. Liu, W. Y. Ma, Mining user similarity based on location history, in: ACM Sigspatial International Conference on Advances in Geographic Information Systems, 2008.
- [35] C. Ying, T. Xu, Design, analysis, and implementation of a large-scale realtime location-based information sharing system, in: International Conference on Mobile Systems, 2008.
- [36] D. Lian, X. Xie, F. Zhang, N. J. Yuan, T. Zhou, Y. Rui, B. Data, Mining location-based social networks: A predictive perspective., IEEE Data Eng. Bull. 38 (2) (2015) 35-46.
- [37] M. Ye, P. Yin, W.-C. Lee, Location recommendation for location-based social networks, in: Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems, 2010, pp. 458-461.
- [38] Y. Liu, W. Wei, A. Sun, C. Miao, Exploiting geographical neighborhood characteristics for location recommendation, in: Proceedings of the 23<sup>rd</sup> ACM International Conference on Information and Knowledge Management, ACM, 2014, pp. 739-748.
- [39] Y. Liu, T.-A. N. Pham, G. Cong, Q. Yuan, An experimental evaluation of point-of-interest recommendation in location-based social networks, Proceedings of the VLDB Endowment 10 (10) (2017) 1010-1021.
- [40] H. Wang, M. Terrovitis, N. Mamoulis, Location recommendation in location-based social networks using user check-in data, in: Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 2013, pp. 374-383.
- [41] J. Wang, Z. Wang, D. Zhang, and J. Yan, "Combining knowledge with deep convolutional neural networks for short text classification." in IJCAI, 2017, pp. 2915–2921.

- [42] P. Li, Z. Wang, Z. Ren, L. Bing, and W. Lam, “Neural rating regression with abstractive tips generation for recommendation,” in Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval, 2017, pp. 345–354.
- [43] T. Zhou, Z. Kuscsik, J.-G. Liu, M. Medo, J. R. Wakeling, Y.-C. Zhang, Solving the apparent diversity-accuracy dilemma of recommender systems, Proceedings of the National Academy of Sciences 107 (10) (2010) 4511-4515.
- [44] T. Zhou, J. Ren, M. Medo, Y.-C. Zhang, Bipartite network projection and personal recommendation, Physical Review E 76 (4) (2007) 046115.
- [45] Y.-C. Zhang, M. Blattner, Y.-K. Yu, Heat conduction process on community networks as a recommendation model, Physical Review Letters 99 (15) (2007) 154301.
- [46] M. Ye, P. Yin, W.-C. Lee, D.-L. Lee, Exploiting geographical influence for collaborative point-of-interest recommendation, in: Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval, ACM, 2011, pp. 325-334.

指导教师签字:

2020 年 6 月 29 日

### 硕士学位论文开题论证报告

报告人姓名	张晓洁	年级	2018	专业	计算机科学与技术
论文题目	基于用户点评信息的推荐和餐厅倒闭可解释预测算法的研究				

指导教师意见:

该生对推荐和可解释预测算法相关的知识与理论研究比较透彻，参考了许多的文献资料，具有一定的研究价值。本课题结构合理，内容完整，主要观点突出，是学生学习方向的延续，对于提高学生的能力有利。同意该课题开题。

指导教师（签名）:

2020 年 6 月 29 日

评审小组意见:

该毕业论文选题是推荐和预测算法领域的热点问题，论文选题具有较强的创新性和实际应用价值。论文题目契合主题，较好地展现了论文的核心内容和核心思想。开题报告撰写符合规范要求。提供了必要和完整的文献调研。开题答辩着装正式，语句表达清晰。能够较好地回答评审小组的问题。经评审小组讨论后，一致同意开题。

评审小组负责人（签名）:

2020年6月29日

评审结果:

通过(√)

不通过( )

延期( )

学院审批意见:

学院负责人（签名）:



### 论文工作计划表

姓名		张晓洁	开题报告日期	2020.06.20
论文题目		基于用户点评信息的推荐和餐厅倒闭可解释预测算法的研究		
预计 工作 进度	序号	工作内容及预期目标	起止时间	
	1	文献资料综合分析	2020.03 – 2020.04	
	2	开题报告	2020.06.21	
	3	实验准备、阅读文献、设计算法	2020.06 – 2020.08	

	4	进行实验	2020.08 - 2020.10
	5	向导师、学科组作阶段性研究成果汇报	2020.11
	6	撰写学位论文	2020.12 - 2021.02
	7	研究生在学科组汇报论文撰写情况；导师介绍研究生学习的全面情况和对毕业论文的意见	2021.04
	8	论文完成，申请论文答辩	2021.04
	9	论文答辩	2021.05
备注			