

支持灾难救援的在线空间众包匹配算法

刘俊岭^{1,2}, 吴晴晴^{1,2}, 董珊珊^{1,2}, 孙焕良^{1,2}, 许景科^{1,2,3}

(1. 沈阳建筑大学计算机科学与工程学院, 辽宁 沈阳 110168)

(2. 辽宁省城市建设大数据管理与分析重点实验室, 辽宁 沈阳 110168)

(3. 国家特种计算机工程技术研究中心沈阳分中心, 辽宁 沈阳 110168)

[摘要] 灾难发生后人们常常通过社交媒体平台发布志愿者救援与受灾者求助信息, 从这些数据中抽取求助任务与救援人员信息并对两者进行合理匹配可以为救助提供有效支持. 本文将空间众包技术引入灾难救援领域, 提出支持灾难救援的在线空间众包匹配问题. 利用深度学习分类方法与大规模语言模型构建灾难事件信息抽取模型, 实现了救援和求助信息的准确抽取; 设计了任务等级评定方法与动态损失度量, 以反映任务的紧急性和损失的动态变化; 基于动态损失度量提出了一种综合抢占与延迟策略的贪心算法. 通过真实数据集及合成数据集进行详细的实验分析, 与现有算法相比, 提出的综合抢占与延迟的贪心算法总损失至少减少 35%, 验证了所提算法的有效性.

[关键词] 空间众包, 灾难救援, 任务匹配, 任务等级, 最小损失

[中图分类号] TP399 **[文献标志码]** A **[文章编号]** 1001-4616(2024)04-0021-10

Online Spatial Crowdsourcing Matching Algorithm for Disaster Relief

Liu Junling^{1,2}, Wu Qingqing^{1,2}, Dong Shanshan^{1,2}, Sun Huanliang^{1,2}, Xu Jingke^{1,2,3}

(1. School of Computer Science and Engineering, Shenyang Jianzhu University, Shenyang 110168, China)

(2. Liaoning Province Big Data Management and Analysis Laboratory of Urban Construction, Shenyang 110168, China)

(3. Shenyang Branch of National Special Computer Engineering Technology Research Center, Shenyang 110168, China)

Abstract: After disasters, people often post information about volunteer rescue efforts and requests for help from the affected on social media platforms. Extracting the information of help task and rescue personnel from these data and making a reasonable match between them can provide effective support for rescue. In this paper, spatial crowdsourcing technology is introduced into the field of disaster relief, and online spatial crowdsourcing matching problem for disaster relief is proposed. The disaster event information extraction model is constructed by using deep learning classification method and large-scale language model to realize the accurate extraction of rescue and help information. The task rating method and dynamic loss measurement are designed to reflect the urgency of the task and the dynamic change of the loss. A greedy algorithm combining preempt and delay strategies is proposed based on dynamic loss measurement. Through detailed experimental analysis of real data sets and synthetic data sets, the total loss of the greedy algorithm combining preempt and delay strategies is reduced by at least 35% compared with the existing algorithm, and the effectiveness of the proposed algorithm is verified.

Key words: spatial crowdsourcing, disaster relief, task matching, task level, minimum loss

自然灾害如台风、暴雨等会引起人员伤亡与财产损失, 及时的救援至关重要. 由于响应时间和救援人员的限制, 不得不在有限的时间和救援人员的基础上做出最优分配, 以减少损失、拯救生命^[1-2].

通常应急救援涉及大量的任务, 实施救援的过程中要坚持将及时处理紧急任务以降低总损失作为基本原则. 救援任务应区分出轻重缓急, 把人员的安全放在首位. 现有的救援方式以政府部门组织性救援为主. 近年来社交媒体平台(如微博、论坛等)为救援信息的发布提供了新的渠道, 这些平台可以同时发布救助与求助两类信息^[3-4]. 例如, 2021 年 7 月河南遭受重大水灾, 由于受灾范围大, 一些灾害区域难以得到及

收稿日期: 2024-05-30.

基金项目: 国家自然科学基金项目(62073227)、国家重点研发计划课题(2021YFF0306303)、辽宁省教育厅资助项目(JYTMS20231596、LJZ2021008).

通讯作者: 孙焕良, 博士, 教授, 博士生导师, 研究方向: 空间数据管理、数据挖掘等. E-mail: sunhl@sjzu.edu.cn

时救援。随后在平台中出现了“救命文档”,该文档登记了受灾求助情况及可以提供援助的相关信息。但事出紧急且信息量较大,这种由志愿者各自查询并进行救助的方式无法有效减少灾难所带来的损失。

基于以上观察,本文将空间众包技术引入灾难救援领域,提出了支持灾难救援的在线空间众包匹配算法,利用空间众包技术有效地将求助任务与救援人员进行匹配,以最大限度地减少灾害损失。

现有的空间众包应用领域大多属于商业服务类,优化目标通常为工人收益最大化或总体任务分配最大化^[5-6]。然而,灾难救援应用中应以损失最小化为目标,且灾难救援应用中损失会随救援的延误而增大,因此利用现有空间众包算法难以直接有效求解灾难救援的空间众包问题。

灾难救援应用中空间众包问题存在的挑战包括:①有效地抽取求助任务与救援人员信息;②自动评定其紧急程度和等级,设计合理的动态损失度量;③考虑时间、地点并结合动态损失度量,实现实时的最小损失分配。其中,当空间众包应用涉及从多源非结构化数据中抽取任务与救援人员信息时均存在挑战①和②,挑战③是灾难救援空间众包问题所特有的。

为了应对以上挑战,首先,基于 TencentPretrain 框架对大规模语言预训练模型进行指令微调,设计了以水灾求助为例的信息抽取模型,实现准确、自动抽取救援信息;其次,使用 FAHP 和 CRITIC 主客观结合方法评定救援信息的等级,设计动态损失度量;最后,提出了抢占与延迟策略,结合两种策略设计一种任务匹配算法,旨在优化资源分配并最小化整体损失。实验结果证明综合抢占与延迟策略的贪心算法可以更加有效地减少总损失。本文提出的空间众包解决方案可以应用于任何以损失最小为优化目标的众包应用场景。

1 相关工作

1.1 信息抽取

在自然灾害发生后,整合社交媒体发布的救援相关信息,提取出空间任务和救援人员的基本信息,从而为后续任务分配奠定基础。信息抽取的主要任务是从自然语言文本中抽取特定的事实信息。

信息抽取方法分为传统的基于规则及统计的方法^[7]、基于机器学习与深度学习的方法^[8]。随着机器学习和深度学习迅速发展,学者们研究了使用深度学习模型进行实体抽取^[9-10]。目前大规模预训练语言模型凭借其强大的语言理解和上下文处理能力,在实体识别方面的应用中展现出了良好的性能^[11-12]。

为了在灾难期间从媒体信息中提取有效数据,本文使用 ChatFlow 大语言模型构建了社交媒体数据中以水灾事件为例的救援信息提取模型。

1.2 空间众包

Kazemi 等^[5]在 2012 年提出了空间众包概念,不同于传统众包,空间众包是依托移动设备、结合互联网平台、利用人群的潜力来执行具有空间性质的任务。

空间众包的核心为工人与任务的匹配^[13],匹配的目标为最大化匹配收益,收益包括分配任务数量、工作人员奖励、平台总体效用等^[14]。根据应用场景不同,空间众包可分为离线空间众包^[15]与在线空间众包^[16]。这些模型旨在在线状态下最大化完成任务数量和整体效用^[17]。本文研究的灾难救援空间众包是在线场景下的空间众包。

近年来空间众包在灾难救援领域的应用备受关注。陶阳阳^[18]列出了近年来众包模式下志愿者参与灾难救援的案例,并分析了志愿者对众包模式下参与救援行动的意愿。此外,文献[19-20]针对众包视角下救灾减灾的社会参与机制进行了研究,并提出了有效的众包模式下救灾减灾方法。这些研究从理论层面证明了空间众包模式在灾难救援领域的可行性。

1.3 灾难救援相关信息技术

信息技术在灾难救援中起着重要的作用,全国应急避难场所综合信息管理服务系统等承担着协调救援任务、发布灾情信息和协调志愿者资源等重要职责。

此外,近年来一些非官方的信息技术应用也在救援中发挥着重要作用。例如,2021 年 7 月 20 日,郑州特大暴雨事件中使用共享在线文档创建了“救命文档”^[21]。Ushahidi 是一个典型的开源救援平台,为全球范围受灾人员与救援组织提供联系渠道。OpenStreetMap 和 GoogleMaps 地图服务平台也提供用户实时灾情信息上传的功能^[22]。

本文研究旨在抽取公共平台发布的灾难救援信息,将空间众包技术引入灾难救援,为志愿者参与救援

工作提供任务匹配指导,最大化地降低灾难所带来的损失.

2 相关定义

首先介绍使用的相关定义,然后引出本文提出的空间众包匹配问题定义.

定义 1 空间任务. 空间任务表示为 $s(i, l, t_0, t_{\max}, r, n)$, 其中 i 为任务编号, l 为任务所在位置, t_0 为任务请求时间, t_{\max} 表示此任务的截止时间, r 为任务等级, n 为任务所需人员的数量.

任务等级 r 表示任务 s 的紧急程度, 涉及人身安全的任务等级最高, 本文将任务等级 r 设为 1~5 级, 其中 1 级最高. 截止时间 t_{\max} 表示任务的最长等待时间, 即过了此时间, 任务等级随之变高.

定义 2 空间任务损失. 给定一个空间任务 s , 任务请求开始所经历的时间 t , 空间任务 s 的损失表示为 $L(s, t)$, 其值会随时间增长而增大. 当 $t \geq t_{\max}$ 时, 任务 s 损失达到最大值, 表示为 $L(s, t_{\max})$. 不同等级的空间任务对应的损失变化率不同, 等级越高的任务损失越大.

定义 3 救援人员. 救援人员表示为 $w(i, l, t_0)$, 其中 i 为救援人员编号, l 表示人员 w 所在地点, t_0 为其请求时间. 救援人员简称为人员, 只有当人员空闲时, 才能为其分配任务.

定义 4 最小损失任务匹配问题. 给定空间任务序列 $S = \{s_1, s_2, \dots\}$, 其中 s_i 的请求时间 $< s_j$ 的请求时间 ($i < j$), 救援人员序列 $W = \{w_1, w_2, \dots\}$, 其中 w_i 的请求时间 $< w_j$ 的请求时间 ($i < j$), 连续等间隔时间段集合 $T = \{T_1, T_2, \dots, T_n\}$, 最小损失任务匹配问题指在 T_i 时间段上根据任务等级及任务所需人员数量将人员分配给任务, 表示为 $M(S, W) = \{< s_1, W_1 >, < s_2, W_2 >, \dots\}$, 其中 $< s_i, W_i >$ 指任务 s_i 的匹配. 匹配 $M(S, W)$ 的损失为 $L(M(S, W)) = \sum L(< s_i, W_i >)$. 最小损失任务匹配表示为 $M_{\min}(S, W)$, 是所有匹配中损失最小的匹配, 满足 $L(M_{\min}(S, W)) = \min L(M_j(S, W))$, 其中 $M_j(S, W)$ 为任一匹配.

可将最小损失任务匹配问题归约为 0-1 背包问题, 从而证明该问题是 NP 难问题.

定义 5 抢占. 给定一个高等级任务 s_j , 当空闲人员序列为空时, 对于任务 s_i 已分配的结果 $< s_i, W_i >$, 如果 $(L(s_j, t_0) - L(s_j, t)) > L(s_i, t_0)$, 则将 W_i 重新分配给 s_j , 称为任务 s_j 抢占任务 s_i . 其中, $L(s_i, t_0)$ 与 $L(s_j, t_0)$ 为任务 s_i 与 s_j 的最大损失, $L(s_j, t)$ 为分配后的损失, t 为再分配的人员到达任务 s_j 的时间. $L(s_j, t_0) - L(s_j, t)$ 称为抢占收益, $L(s_i, t_0)$ 为抢占代价.

定义 6 延迟. 任务按时间段分批匹配, 当救援人员不足时, 低等级任务延迟到下一批处理队列中, 高等级任务立即匹配. 每个任务 s 有一个记时器 t_d , 其中 t_d 等于任务 s 当前时间 t_{now} 减去任务请求时间 t_0 . 每延迟一次则放入下一批处理队列, 下一批处理时综合考虑时间与任务等级, 定义任务优先级因子为 $p_d(s)$:

$$p_d(s) = \alpha \times s_r + \beta \times s_{t_d}, \quad (1)$$

式中, $\alpha + \beta = 1$, 且 $\alpha \geq \beta$.

表 1 和表 2 给出了不同时间段到达的任务与人员示例. 假定每个时间段为 0.5 h, 设置人员移动速度为 4. 救援人员和任务按表 1、2 的时间段实时出现, 任务等级由高到低表示为红色、橙色、黄色、绿色、蓝色.

表 1 任务示例

Table 1 Example of tasks

T	s_i	s_l	s_r	s_n	$L(s, t_{\max})$	$s_{t_{\max}}/h$
T_1	s_1	(1, 1)	5	2	10	4
T_1	s_2	(2, 5)	4	2	20	4
T_2	s_3	(3, 3)	3	1	40	4
T_2	s_4	(6, 2)	1	3	160	4
T_3	s_5	(7, 1)	1	2	160	4
T_3	s_6	(3, 6)	2	3	80	4

表 2 救援人员示例

Table 2 Example of workers

T	w_i	w_l
T_1	w_1, w_2, w_3, w_4	(2, 2), (1, 6), (2, 7), (3, 1)
T_3	w_5, w_6, w_7, w_8	(5, 1), (5, 6), (5, 4), (7, 4)

图 1 为采用文献[6]中提出的最近邻优先算法(nearest neighbor priority algorithm, NNP)的匹配结果: $\{< s_1, \{w_1, w_4\} >, < s_2, \{w_2, w_3\} >, < s_3, \{w_7\} >, < s_4, \{w_5, w_6, w_8\} >\}$. 由于高等级任务 s_4 未在时间段 T_2 内及时匹配, 直到时间段 T_3 才匹配成功, 匹配的总损失为 309.38.

图 2 为最近邻优先算法中加入任务等级属性的匹配结果: $\{< s_1, \{w_1, w_4\} >, < s_2, \{w_2, w_3\} >, < s_3, \{w_6\} >, < s_4, \{w_5, w_7, w_8\} >\}$. 此匹配的总损失为 302.98. 与图 1 中匹配相比, 任务 s_4 获得了更近距离的人员匹配, 且匹配总距离更小.

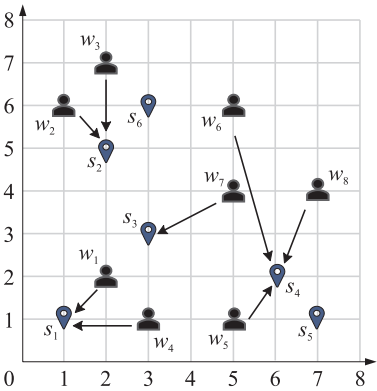


图 1 最近邻贪心算法匹配示例

Fig. 1 Matching example of nearest neighbor greedy algorithm

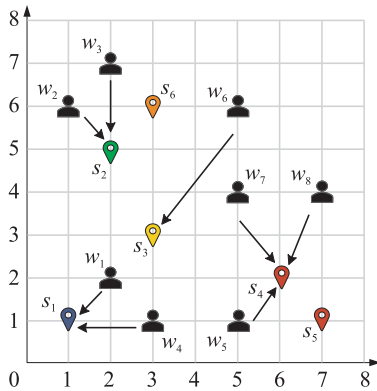


图 2 结合任务等级的最近邻贪心算法匹配示例

Fig. 2 Matching example of nearest neighbor greedy algorithm combining task levels

3 匹配算法

本文提出的支持灾难救援的在线空间众包匹配算法模型如图 3 所示. 该模型包括信息抽取模型构建、任务等级评定与动态损失度量设计、任务匹配算法三部分.

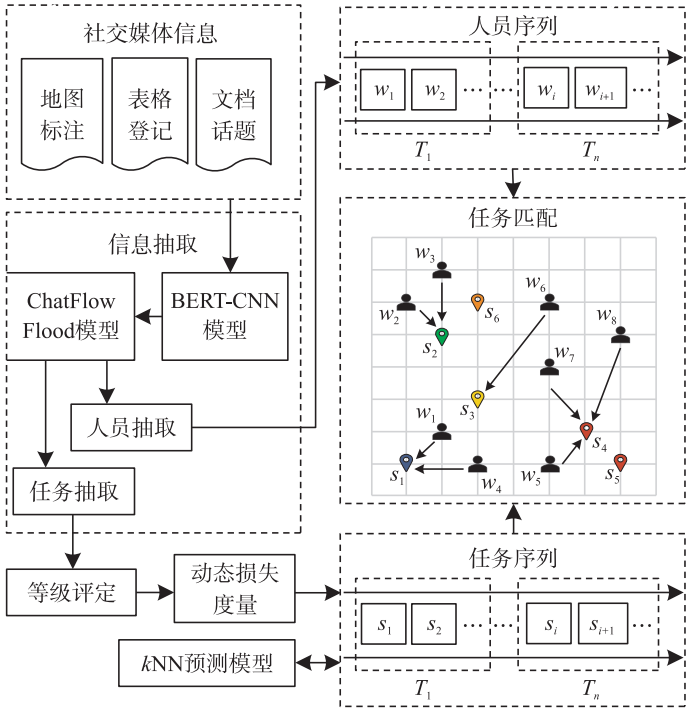


图 3 匹配算法模型

Fig. 3 Matching algorithm model

灾难发生后社交媒体会以不同的形式发布求助信息与救援信息. 首先,采用 BERT-CNN 分类模型将收集到的信息进行分类. 然后,利用设计的 ChatFlowFlood 大语言模型进行任务与人员信息抽取. 对抽取的任务信息进行等级评定及动态损失度量设计. 最后,结合 k NN 预测模型进行任务预测,将得到的任务序列与人员按照所设计的匹配算法进行在线匹配.

3.1 任务与人员信息抽取

针对水灾事件,本文提出了社交媒体数据中水灾事件求助、救助信息提取模型. 该模型由水灾事件语料库构建、基于 BERT-CNN 模型的自动分类、ChatFlowFlood 信息抽取模型 3 个部分构成.

首先,在水灾事件语料库构建部分,通过爬取微博河南暴雨互助话题中水灾事件相关信息获得原始数据集,利用数据集对水灾事件知识体系进行构建. 结合构建的知识体系将原始数据集进行分类标注得到分类数

据集. 然后,采用 BERT-CNN 训练分类数据,并预测出求助信息. 最后,将求助信息结合水灾事件知识体系通过 GPT-4 标注数据得到指令微调数据集. 基于 ChatFlow 模型进行指令微调得到信息抽取模型 ChatFlowFlood,通过模型进行推理,得到结构化的求助需求信息. 对于抽取出的信息的真实性与一致性采用文献[23]的处理方法进行校验.

3.2 任务等级评定及动态损失度量

灾害发生后需要迅速做出反应,来减少总损失. 在此期间应尽可能保证被困人员的生命安全,减少人员伤亡,因此需要将求助信息评级,使紧迫的求助更快地得到救助. 借鉴陈报章等^[24]的研究将灾害危险度由高到低划分为 5 个等级,用数字表示为 1、2、3、4、5. 采用主客观综合赋权法对任务的紧急程度进行评定^[25],然后进行模糊综合评价,最终评定出每个任务的等级.

首先按照收集的数据将不同的求助关键词赋予内容和权重,权重越高则越紧急. 不同关键词的灾损指数如表 3 所示,其中 c_1 = 人的损伤程度; c_2 = 急救时医院医护人员不足且急需药品; c_3 = 居住者流失的影响; c_4 = 任务中食品原料不足的影响; c_5 = 不卫生状况的影响.

根据归一化原理,将不同关键词的灾损指数按照等间距原则,在 0~1 之间对灾害的危险度进行赋值,然后通过序关系法主观得到权值,通过 CRTIC 权重法客观确认权值. 本文采用主客观相结合方法对任务紧急程度进行评定,以期获得准确的任务等级评估结果.

灾害发生会产生自身的经济上的损失,救援不及时和人员的移动会加大损失量,即后续损失. 当受灾点得到及时的救助,认为后续损失为给定的初始损失 c . 根据李超超等^[26]的相关研究可知后续损失符合 S 型曲线变化,因此本文采用 S 型损失增长函数 Logistic 进行建模. 给定任务 s ,其损失函数为:

$$L(s, t) = \frac{A}{1 + \left(\frac{A}{c} - 1 \right) e^{f(s_r)t}}, \quad (2)$$

式中,参数 A 为最大损失饱和值; c 为初始损失; k 为曲线的变化率,其值与任务等级相关,即 $k = f(s_r)$.

函数自变量 t 为任务请求开始所经历的时间,此时间需要对实际时间进行最大-最小规范化处理. 表 4 给出了 1~5 级任务损失函数的参数取值.

3.3 综合抢占与延迟策略的贪心算法

支持灾难救援的在线空间众包匹配问题是 NP-hard 问题,解决此类问题通常采用贪心算法. 本文借鉴文献[5]处理在线空间众包问题算法,提出了考虑任务

等级与最近邻的基本贪心算法. 但是贪心算法只在同一时间段内实现按等级与距离进行优化匹配,得到的往往是局部有效解,不是全局最优解,因此在贪心算法的基础上提出了两种启发式策略.

如果当前时间段出现高等级的任务、且人员数不充分时,则会延误高等级任务的分配,这将增加整体损失. 因此,本文提出了抢占策略,在定义 5 中给出了抢占的定义. 抢占策略是指当人员数量不足时,重新将上一时段已分配给低等级任务的救援人员再分配给后到来的高等级任务. 抢占行为也会产生一定的代价,低等级任务会因得不到匹配产生损失. 因此,需要在抢占时平衡收益和代价,确保抢占的收益大于代价.

由于高等级任务可能在低等级任务后到达,为了减少抢占次数,本文提出了延迟策略,在定义 6 中给出了延迟的定义. 延迟策略是指直接将低等级任务延迟匹配,预先留取一部分空闲人员给后续到来的更紧急的高等级任务,以降低整体的损失.

抢占与延迟策略各自有适用情景,当任务与救援人员大量出现,救援人员数量不足且高等级任务较多时,需要同时执行抢占与延迟策略,使更多的高等级任务及时得到匹配. 因此,本文提出了综合抢占与延迟策略的贪心算法,以适用于各种场景下的任务匹配,最小化整体损失.

综合抢占与延迟策略的贪心算法如算法 1 所示. 给定任务集 S 和人员集 W ,对于每个任务请求 s ,如果为

表 3 不同关键词权重

Table 3 Different keyword weights

任务等级	灾损指数	求助关键词	损失程度
1	0.9	c_1	高
2	0.7	c_2	较高
3	0.5	c_3	中等
4	0.3	c_4	较低
5	0.1	c_5	低

表 4 损失函数参数设置

Table 4 Loss function parameter settings

s_r	c	A	$f(s_r)$
1	20	160	-0.5
2	10	80	-0.4
3	8	40	-0.3
4	5	20	-0.2
5	3	10	-0.1

高等级任务且人员充足则直接匹配人员(第 6~9 行),若人员数量不足则按照抢占策略抢占低等级任务已匹配的人员(第 10~13 行),抢占不成功则加入任务池(第 14 行);如果为低等级任务则先判断是否需要延迟,不需要则直接进行匹配(第 15~18 行),否则进行延迟(第 19~20 行). 对于人员请求,将其分配给优先度最高的任务(第 21~27 行). 最终返回任务匹配结果 $M(S, W)$.

算法 1 Preempt-Delay

```

Input: 任务集  $S$ , 救援人员集  $W$ , 预测空间任务集  $S'$ 
Output: 任务分配  $M(S, W)$ 
1  $M(S, W) = \emptyset$ ;
2 Predicting  $S'$  using  $kNN$ ;
3 for(each new arrival request) do
4   if(the request is a task  $s$ ) then
5     if( $s_r < r'$ ) then
6       if(freeworkers is not empty) then
7         if( $n$  workers closest to  $\text{dist}(w[i]_l, s_l)$ ) then
8           Append  $n * \{w[i], s\}$  to  $\langle s, W \rangle$ ;
9           freeworkers = freeworkers -  $\{w[i]\}$ ;
10        else if(busyworkers is not empty) then
11          if( $((L(s, t_0) - L(s, t)) > L(s', t_0)) \&\& n$  workers closest to  $\text{dist}(w'[j]_l, s_l)$ ) then
12            Append  $n * \{w'[j], s\}$  to  $\langle s, W \rangle$ ;
13            taskpool = taskpool +  $\{s'\}$ ;
14          else Append  $s$  to taskpool;
15        else if( $s_r \geq r' \&\& S'_{T_i} = 0$ ) then
16          if( $n$  workers closest to  $\text{dist}(w[i]_l, s_l)$ ) then
17            Append  $n * \{w[i], s\}$  to  $\langle s, W \rangle$ ;
18            freeworkers = freeworkers -  $\{w[i]\}$ ;
19          else Append  $s$  to taskpool;
20           $S'_{T_i} = S'_{T_i} - s_n$ ;
21        else if(the request is a worker  $w$ ) then
22          if( $(p_d(s[j]) \text{ is max } \&\& (t_{\text{now}} + \text{dist}(w_l, s[j]_l)) \leq s[j]_{t_{\text{max}}})$ ) then
23             $\langle s, W \rangle = \langle s, W \rangle + \{w, s[j]\}$ ;
24            if(the workers of  $s[j]$  is full) then
25              taskpool = taskpool -  $\{s[j]\}$ ;
26            Append  $\langle s, W \rangle$  to  $M(S, W)$ ;
27          else Append  $w$  to freeworkers;
28 return  $M(S, W)$ ;
    
```

由于延迟策略是基于后续时间段会有高等级的任务出现这一假设下提出的,因此在处理每时间段任务前要预测下一时间段可能的高等级任务数量. 由于所需预测的任务等级具有时空关联性,因此借鉴文献[27]设计了一种基于 kNN 的任务等级预测模型. 针对水灾事件,根据任务位置的前 5 h 降雨量及所处区域风险等级进行任务等级预测. 通过该模型得到任务等级预测集合 S' (算法 1 第 2 行), S'_{T_i} 为 S' 中 T_i 时刻的高等级任务数,以此来控制低等级任务延迟的数量.

算法 1 的时间复杂度为 $O(|S| * |W|)$, 其中 $|S|$ 为任务数, $|W|$ 为人员数. 图 4 为 Preempt-Delay 算法的匹配结果. 在 T_1, T_2 时间段由于预测到后续有高等级任务要到来,因此延迟低等级任务 s_2, s_3, s_4 的匹配. 在 T_2 和 T_3 时间段成功为任

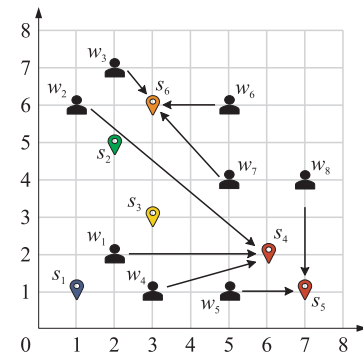


图 4 综合抢占与延迟策略的贪心算法匹配示例
Fig. 4 Matching example of greedy algorithm combining preempt and delay strategies

务 s_1, s_5, s_6 匹配上所需要的工人. 因此, 结合抢占与延迟策略后所有的高等级任务均被匹配, 总损失最低为 148.83, 匹配结果为 $\{<s_4, \{w_1, w_2, w_4\}>, <s_5, \{w_5, w_8\}>, <s_6, \{w_3, w_6, w_7\}>\}$.

4 实验评价

4.1 数据集及参数设置

实验数据来自 2021 年“7·20 河南暴雨”事件后社交媒体上的求助及援助信息, 共 4 063 条. 经过信息抽取模型处理后得到 2 173 条任务数据. 由于实际数据中救援人员数量不足, 通过合成数据的方式进行了扩充和增补, 生成的任务等待时间遵循高斯分布, 人员释放时间遵循泊松分布.

实验是在配备 Intel(R) Core(TM) i7-7820X CPU@3.60 GHz 和 64 GB 的机器上进行的, 采用的编程语言为 C++, 表 5、6 分别展示了实验、最大损失参数设置, 其中所有参数的默认值加粗表示.

可抢占等级参数设置包括 5 种不同的可抢占等级数: 0 表示所有任务均不可以抢占, 1 表示仅允许 1 等级任务可以抢占比其等级低的任务, 2 表示允许 1、2 等级任务抢占, 随后的实验组的抢占级别依次增加. 可延迟等级参数设置包括 5 种不同的延迟等级数: 0 表示所有任务均未被延迟, 1 表示只延迟 5 等级任务, 2 表示延迟 4、5 等级任务, 3 表示延迟 3~5 等级任务, 4 表示延迟 2~5 等级任务.

表 5 实验参数设置

Table 5 Experimental parameter settings

参数名	参数值
任务数 $ S $	500, 1 000 , 1 500, 2 000, 2 500
人员数 $ W $	1 000, 2 000, 3 000 , 4 000, 5 000
任务等级	1, 2, 3, 4, 5
时间段数量 $ T $	10

表 6 最大损失参数设置

Table 6 Maxloss parameter settings

等级	1	2	3	4	5
	50	40	30	20	10
	160	80	40	20	10
最大损失 $ A $	210	130	70	30	10
	310	190	100	40	10
	410	250	130	50	10

针对所提出的支持灾难救援的在线空间众包匹配问题, 对基准算法与本文提出的综合抢占与延迟策略的匹配算法进行比较, 比较度量为匹配的总损失 L . 具体的算法如下:

(1) NNP: 最近邻优先的贪心算法. 本文使用现有空间众包研究中的基本贪心算法作为基本算法^[28], 该算法优先选择距离最近的人员进行任务匹配.

(2) NNGP: 结合任务等级的最近邻优先贪心算法. 在算法 NNP 的基础上, 借鉴文献[5]中收益最大化思想, 并结合任务等级设计的贪心算法.

(3) Preempt: 结合抢占策略的贪心算法. 在任务匹配时, 高等级任务可以抢占低等级任务已分配的人员, 将算法 1 去除延迟策略可以得到 Preempt 算法.

(4) Delay: 结合延迟策略的贪心算法. 任务匹配过程中, 低等级任务按照下一时间段预测高等级任务情况进行延迟操作, 将算法 1 去除抢占策略可以得到 Delay 算法.

(5) Preempt-Delay: 综合抢占与延迟的贪心算法. Preempt-Delay 集成了抢占与延迟两种策略, 适合应用于各种场景下的任务匹配.

其中 NNP 与 NNGP 为现有算法, Preempt、Delay、Preempt-Delay 算法为本文提出的算法.

4.2 实验结果

4.2.1 任务数量对总损失的影响

图 5 给出了 5 种算法在不同任务数量下总损失的变化情况. 从图 5 可以看出, 在人员数量一定的情况下, 随着任务数量的增加, 5 种算法的损失均呈上升趋势. 在任务数量较少时, 5 种算法的损失都相对较小. 随着任务数量的增加, Delay、Preempt-Delay 算法的优势更为显著, 这是由于在预测情况下将低等级任务延迟, 以便匹配更多的高等级任务.

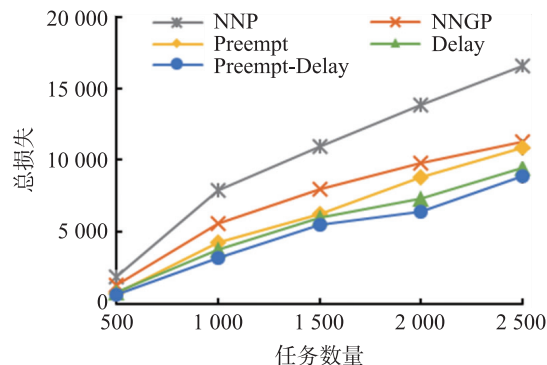


图 5 任务数量对总损失的影响

Fig. 5 Effect of number of tasks on total loss

4.2.2 人员数量对总损失的影响

图 6 给出了 5 种算法在不同人员数量下总损失的变化情况. 从图 6 可以看出,在任务数量一定的情况下,随着人员数量的增加,成功匹配的任务数量增加,从而降低了总损失. 在人员数量较少时,5 种算法都表现出较大的损失. 但随着人员数量增加至一定水平时,NNP 算法和 NNGP 算法的损失显著减少. Preempt 算法、Delay 算法和 Preempt-Delay 算法的损失逐渐减小并趋于稳定. 这是因为随着人员数量的增加,在抢占和延迟策略下,高等级任务被优先匹配完成,剩余低等级任务再被匹配,导致损失不再显著降低. 因此,当人员数量不足时抢占与延迟策略具有更明显的优势.

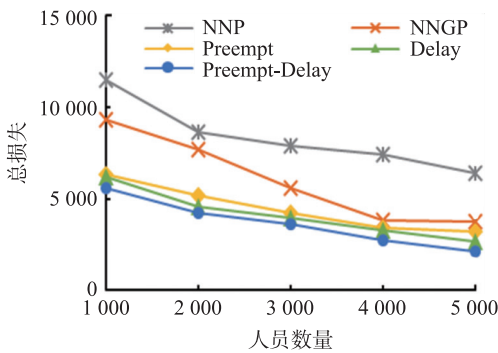


图 6 人员数量对总损失的影响

Fig. 6 Effect of number of workers on total loss

4.2.3 最大损失值对总损失的影响

图 7 是 5 种匹配算法的总损失随最大损失值变化的对比结果. 从图 7 可以看出,任务的最大损失值越大,总损失会越大. NNP 算法的总损失明显高于其他 4 种算法,这是由于 NNP 算法没有对高等级任务及时匹配使得总损失加大. 虽然最大损失值对高等级任务影响较大,但利用抢占和延迟策略的算法 Preempt、Delay、Preempt-Delay 优先匹配高等级任务,使得总损失增加并不显著.

4.2.4 可抢占等级数对总损失的影响

图 8 给出了不同可抢占等级数对总损失的影响. 从图 8 可以看出,在最高可抢占等级为 2 级时,总损失最小. 对于 NNP、NNGP 和 Delay 算法,由于未引入抢占策略,总损失未发生变化. 相较于 Preempt 算法,Preempt-Delay 算法在最高可抢占等级为 2 级时,总损失减少了 15%.

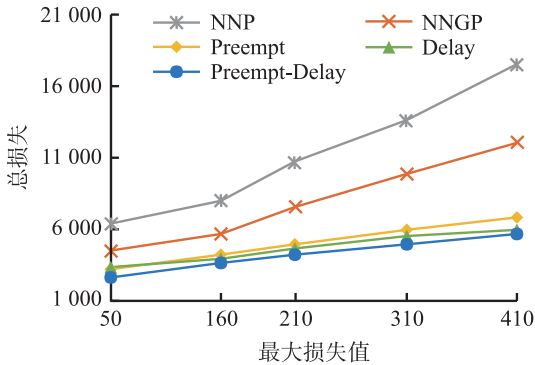


图 7 最大损失值对总损失的影响

Fig. 7 Effect of maximum loss on total loss

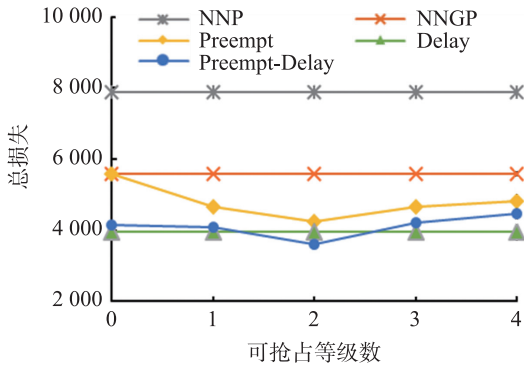


图 8 可抢占等级数对总损失的影响

Fig. 8 Effect of preempt level on total loss

4.2.5 可延迟等级数对总损失的影响

图 9 给出了不同延迟等级数对总损失的影响. 由于 NNP、NNGP 和 Preempt 算法没有采用延迟策略,因此总损失没有变化. Delay 和 Preempt-Delay 算法对不同任务等级的延迟处理产生了不同的效果. 在仅对任务等级较低的 4 级、5 级任务进行延迟操作时,Delay、Preempt-Delay 算法的总损失降到最小.

从以上 5 组实验结果可以得出,与 NNP、NNGP 两种现有基准算法相比,所提出的综合抢占与延迟的贪心算法 Preempt-Delay 在参数取默认值时总损失分别降低了 54%、35%.

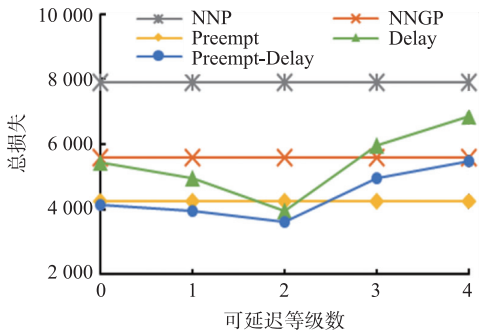


图 9 可延迟等级数对总损失的影响

Fig. 9 Effect of delay level on total loss

5 结论

本文提出了一种支持灾难救援的在线空间众包解决方案,旨在实现灾后任务匹配损失最小化. 基于大规模语言预训练模型构建了信息抽取模型,实现对社交媒体信息中任务和人员信息的抽取,同时设计救援任务等级评定方法与动态损失度量. 由于支持灾难救援的在线空间众包匹配问题是 NP-hard 问题,因此提出了抢占和延迟策略,并结合两种策略设计了一种任务匹配算法,该算法可以在救援人员数量不足且高等级任务较多的情况下最大程度地减少总损失. 本文在真实数据及合成数据集上进行实验,与现有算法相比,所提出的综合抢占与延迟的贪心算法在总损失方面至少减少了 35%,验证了所提算法的有效性.

[参考文献]

- [1] ZHENG Y J, LING H F, XU X L, et al. Emergency scheduling of engineering rescue tasks in disaster relief operations and its application in China[J]. International transactions in operational research, 2015, 22(3): 503–518.
- [2] MOLLAH A K, SADHUKHAN S, DAS P, et al. A cost optimization model and solutions for shelter allocation and relief distribution in flood scenario[J]. International journal of disaster risk reduction, 2018, 31: 1187–1198.
- [3] BEIGI G, HU X, MACIEJEWSKI R, et al. An overview of sentiment analysis in social media and its applications in disaster relief[M]//PEDRYCZ W, CHEN S. Sentiment Analysis and Ontology Engineering. Cham: Springer, 2016.
- [4] GAO H, BARBIER G, GOOLSBY R. Harnessing the crowdsourcing power of social media for disaster relief[J]. IEEE intelligent systems, 2011, 26(3): 10–14.
- [5] KAZEMI L, SHAHABI C. Geocrowd: enabling query answering with spatial crowdsourcing[C]//Proceedings of the 20th International Conference on Advances in Geographic Information Systems. Redondo Beach: Association for Computing Machinery, 2012: 189–198.
- [6] WU B, HAN K, ZHANG E. On the task assignment with group fairness for spatial crowdsourcing[J]. Information processing & management, 2023, 60(2): 103175–103187.
- [7] GUO X Y, HE T T. Survey about research on information extraction[J]. Computer science, 2015, 42(2): 14–17.
- [8] 王剑, 彭雨琦, 赵宇斐, 等. 基于深度学习的社交网络舆情信息抽取方法综述[J]. 计算机科学, 2022, 49(8): 279–293.
- [9] 张仰森, 刘帅康, 刘洋, 等. 基于深度学习的实体关系联合抽取研究综述[J]. 电子学报, 2023, 51(4): 1093–1116.
- [10] 王浩畅, 周柳莲, MARIUS G P. 基于深度学习的事件抽取研究综述[J]. 软件学报, 2023, 34(8): 3905–3923.
- [11] 车万翔, 窦志成, 冯岩松, 等. 大模型时代的自然语言处理: 挑战、机遇与发展[J]. 中国科学: 信息科学, 2023, 53(9): 1645–1687.
- [12] 邓成汝, 凌捷. 融合预训练模型与神经网络的实体关系抽取[J]. 计算机工程与设计, 2023, 44(7): 2023–2029.
- [13] TONG Y, ZHOU Z, ZENG Y, et al. Spatial crowd-sourcing: a survey[J]. The VLDB journal, 2020, 29: 217–250.
- [14] CHENG P, JIAN X, CHEN L. An experimental evaluation of task assignment in spatial crowdsourcing[J]. Proceedings of the VLDB endowment, 2018, 11(11): 1428–1440.
- [15] DENG D, SHAHABI C, DEMIRYUREK U. Maximizing the number of worker's self-selected tasks in spatial crowdsourcing[C]//Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. Orlando: Association for Computing Machinery, 2013: 314–323.
- [16] DANIEL F, KUCHERBAEV P, CAPPIELLO C, et al. Quality control in crowdsourcing: a survey of quality attributes, assessment techniques, and assurance actions[J]. ACM computing surveys, 2018, 51(1): 1–40.
- [17] DENG D, SHAHABI C, ZHU L. Task matching and scheduling for multiple workers in spatial crowdsourcing[C]//Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems. Seattle: Association of Computing Machinery, 2015: 1–10.
- [18] 陶阳阳. 众包模式下志愿者参与应急管理的意愿及影响因素研究[D]. 成都: 四川大学, 2021.
- [19] 朱国齐, 郑乾, 赵雨晴. 众包视角下救灾减灾的社会参与系统运行机制研究[J]. 价值工程, 2018, 37(33): 286–289.
- [20] HUTAGALUNG S, INDRAJAT H. Developing a crowdsourcing-based disaster relief model based on public participation[J]. International journal of safety and security engineering, 2023, 13(1): 115–120.
- [21] 刘胜枝. 媒介向善中的“救命文档”共享型媒介的特征与启示[J]. 人民论坛, 2022(9): 86–89.
- [22] ZOOK M, GRAHAM M, SHELTON T, et al. Volunteered geographic information and crowdsourcing disaster relief: a case study

- of the Haitian earthquake[J]. World medical & health policy, 2010, 2(2): 7–33.
- [23] 孙焕良,王思懿,刘俊岭,等. 社交媒体数据中水灾事件求助信息提取模型[J]. 计算机应用, 2024, 44(8): 2437–2445.
- [24] 陈报章,仲崇庆. 自然灾害风险损失等级评估的初步研究[J]. 灾害学, 2010, 25(3): 1–5.
- [25] SARMA D, DAS A, DUTTA P, et al. A cost minimization resource allocation model for disaster relief operations with an information crowdsourcing-based MCDM approach[J]. IEEE transactions on engineering management, 2020, 69(5): 2454–2474.
- [26] 李超超,程晓陶,王艳艳,等. 洪涝灾害三参数损失函数的构建 I: 基本原理[J]. 水利学报, 2020, 51(3): 349–357.
- [27] PARADILAGA S N, SULISTYONINGSIH M, LESTARI R K, et al. Flood prediction using inverse distance weighted interpolation of k -nearest neighbor points[C]//2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS. Brussels: IEEE, 2021: 4616–4619.
- [28] CHENG P, JIAN X, CHEN L. Task assignment on spatial crowdsourcing(Technical Report)[J]. arXiv:1605.09675, 2016.

[责任编辑:丁 蓉]