基于Netlogo的社交网络惩罚机制研究

李纪元

2023244125

**摘要**：社交网络的交流氛围是影响用户体验的重要因素，对立、歧视、仇恨言论的存在会极大恶化社交网络的交流氛围。因此，社交网络的惩罚机制能否减少此类言论的产生，是一个很重要的问题。本文利用Netlogo对社交网络进行了建模，并在模型的基础上测试了不同惩罚机制的效果。

**1 引言**

**2.1 问题介绍**

社交网络是当今互联网不可或缺的一部分，根据QUEST MOBILE 2023年的报告，抖音、快手、小红书、哔哩哔哩、微博五大社交网络平台的去重活跃用户规模达到了10.88亿。对于对立、歧视、仇恨言论的治理始终是社交网络运营的主要问题之一，各个社交网络的主要措施是对于此类言论的发言者给予惩罚，比如短时间禁言、封号等等。本文所要探究的主要问题是，哪种惩罚措施对于抑制负面言论的产生最为有效。

**2.2 问题分析**

社交网络的结构可以看作一张有向图，其中节点代表用户，边代表用户之间的消息。用户可能主动发出消息，也可能接收到消息后扩散给其社交圈的其他用户，这个过程可以反复进行，直到消息被遗忘。用户的体验主要由所接收到的消息决定的，一个用户所接收到的正面信息越多，其体验就越好，而负面信息越多，体验就越差。惩罚机制就是针对节点的操作，比如禁止节点创建边（禁言）、或者将节点删除（封号）等。实施不同的惩罚机制，然后观察网络中各个用户的体验良好与否，就可以判断惩罚机制对社交网络交流氛围的影响。

本文所实现的模型已在Github开源：[https://github.com/Nickelth/PunishModel/](https://github.com/Nickelth/PunishModel/。读者)

[读者](https://github.com/Nickelth/PunishModel/。读者)可以将模型导入Netlogo，进行交互并观察结果。

**2 模型设计与说明**

本节将详细介绍本文所构造的社交网络模型。

本文将社交网络建模为一张二维平面上的有向图，用图中的节点代表社交网络中的用户，用图中的有向边表示用户之间发送的消息。节点间的距离表示用户间的联系程度，距离越近，表示用户对同一话题的关注度越高（为了简单，本文的模型认为即使是频繁相互攻击的用户，他们之间的联系也视为紧密）。图1是本文模型的一个例子。

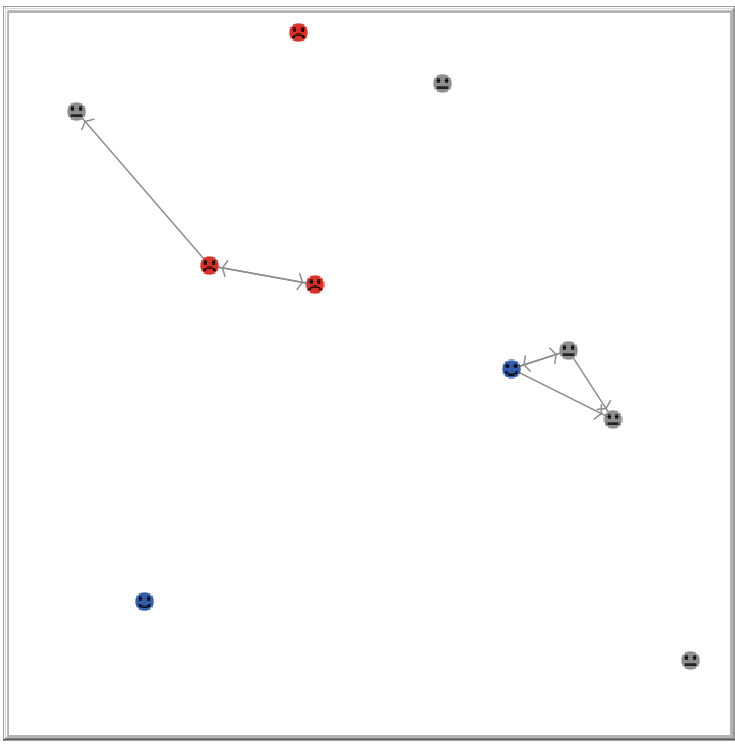


图2.1 社交网络模型示例

在图2.1中，社交网络中有10个用户。每个用户有一个介于0到1之间的心情值，心情值越大，表示该用户心情越好。图2.1中，蓝色图标的是心情值在0.75及以上的用户，红色图标的是心情值在0.25及以下的用户。用户的初始心情值是随机选取的。

本文所设计的模型以迭代方式运行，在每一次迭代的开始，每个用户会以一定概率（本文设置为30%）主动发送消息给周围的用户。消息分为好消息与坏消息两种类型，每种消息有着不同的传播半径，传播半径可以由模型运行者自己设置。每个用户都有影响力数值，每发送一次消息，发送者会根据接收者数量获得一定影响力。为了防止影响力无限增长，本文令影响力每一次迭代都会减弱。影响力越高的用户，所发送的消息传播半径越长。

用户收到消息后，有概率选择转发最新收到的消息，或者遗忘最早收到的消息。每进行一次转发，说明用户对相应的话题感兴趣，所以会向消息的发送者移动一个单位长度（无论该消息是好是坏）。如果用户收到好消息，其心情值会增长，而收到坏消息则会降低。此外，心情值越高的用户，越倾向于发送与扩散好消息；心情值越低的用户，越倾向于发送与扩散坏消息。

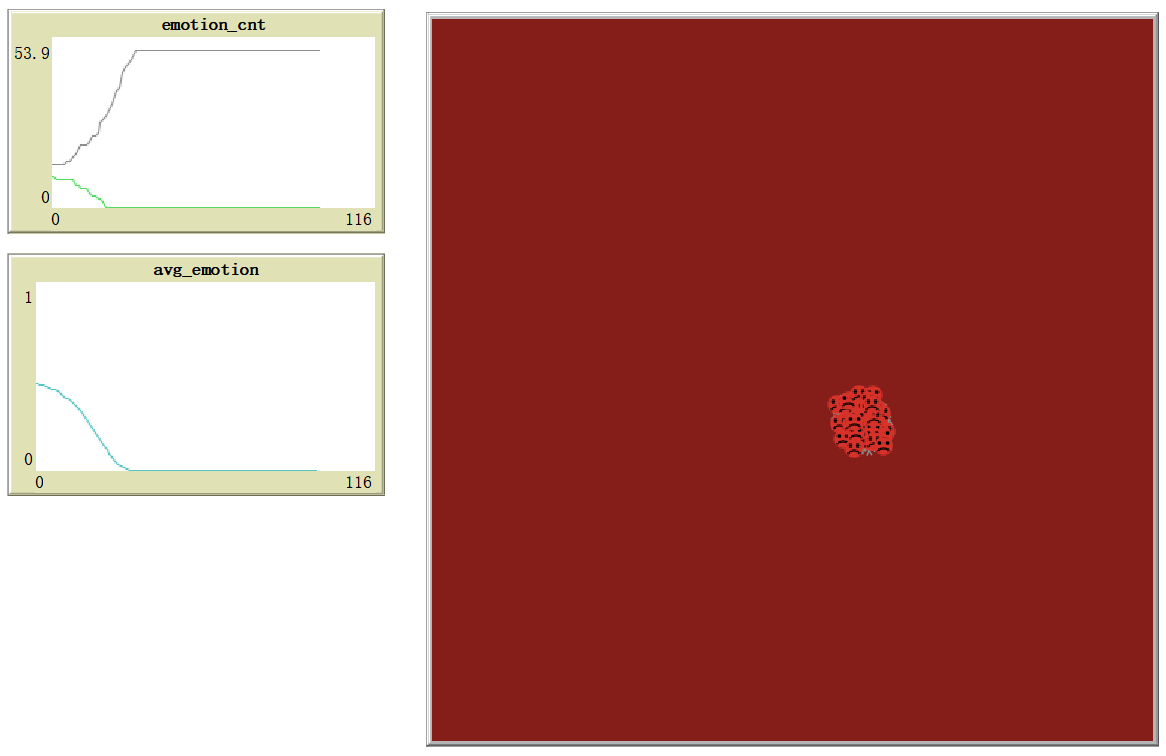


图2.2 模型收敛图示

图2.2是没有惩罚机制、模型运行至收敛的一个例子。用户数量是50，平面大小是18\*18，坏消息的传播半径为10个单位长度，好消息的传播半径为7个单位长度。emotion\_cnt是随时间变化的不同心情人数统计，黑色线表示心情值在0.25及以下的人数，绿色线表示心情值在0.75及以上的人数。avg\_emotion表示所有人的平均心情值。在这个例子中，所有用户最后全都聚集到了一起，无休止地相互攻击下去。

**3 算法设计与说明**

本节将介绍本文的社交网络惩罚机制算法。

在社交网络中，最常用的惩罚手段有禁言（从几天到数个月不等）、封号（删除账户）等。除了发表不良言论的频率以外，本文认为发言者的影响力也是一个影响社交网络交流环境的重要因素。

|  |
| --- |
| 输入：id：用户id  bad\_score：用户发表不良言论次数  influence：用户影响力 |
| punish:  begin  ask user with bad\_score > 5:  set ban\_time = bad\_score + influence  if bad\_score > 0 and bad\_score + influence > 15:  init-user  end |

算法3.1 惩罚机制算法

算法3.1是本文所设计的惩罚机制算法。其过程是，在每一次迭代过程中，令所有不良发言次数（即bad\_score）超过30次的用户，禁言bad\_score + influence轮。如果bad\_score + influence总和超过三十，则进行封禁。简单起见，本文假设社交网络的用户量不变，因此并不直接删除节点，而是令其初始化（刷新各项数值并随机分配到另一个位置）。可以看出，对于发表不良言论次数越多、影响力越大的用户，惩罚的力度越大。

**4 实验与评估**

本节将评估文本所设计的惩罚机制对本文所构造的模型的效果。模型的设置为：平面大小为18\*18，用户数量为50。为了排除随机性，每组实验均重复5次。

首先，将好消息与坏消息的传播半径均设置为4个单位长度。图4.1是其中一次实验的结果。在5次实验中，每一次模型均在300次迭代内收敛，且最终所有用户的心情值均在0.75以上。

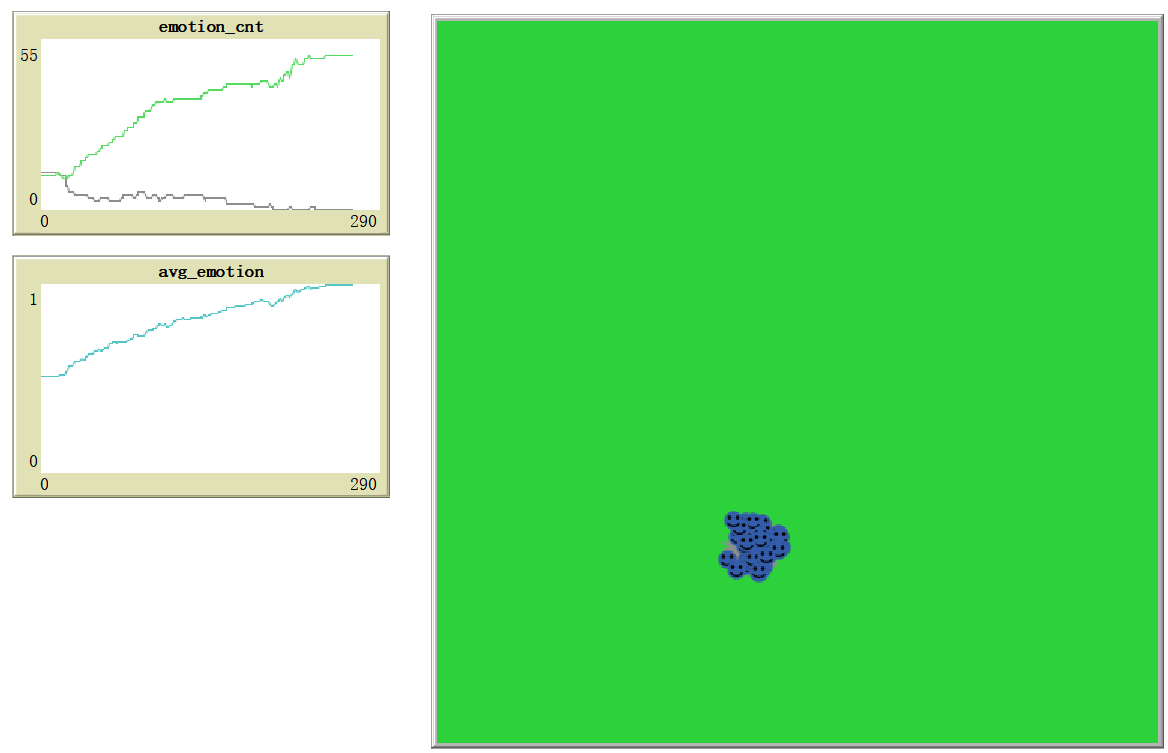


图4.1 好坏消息传播半径相同的实验结果

将坏消息的传播半径更改为好消息的3倍后，模型仍然会在5000次迭代内收敛，如图4.2所示。

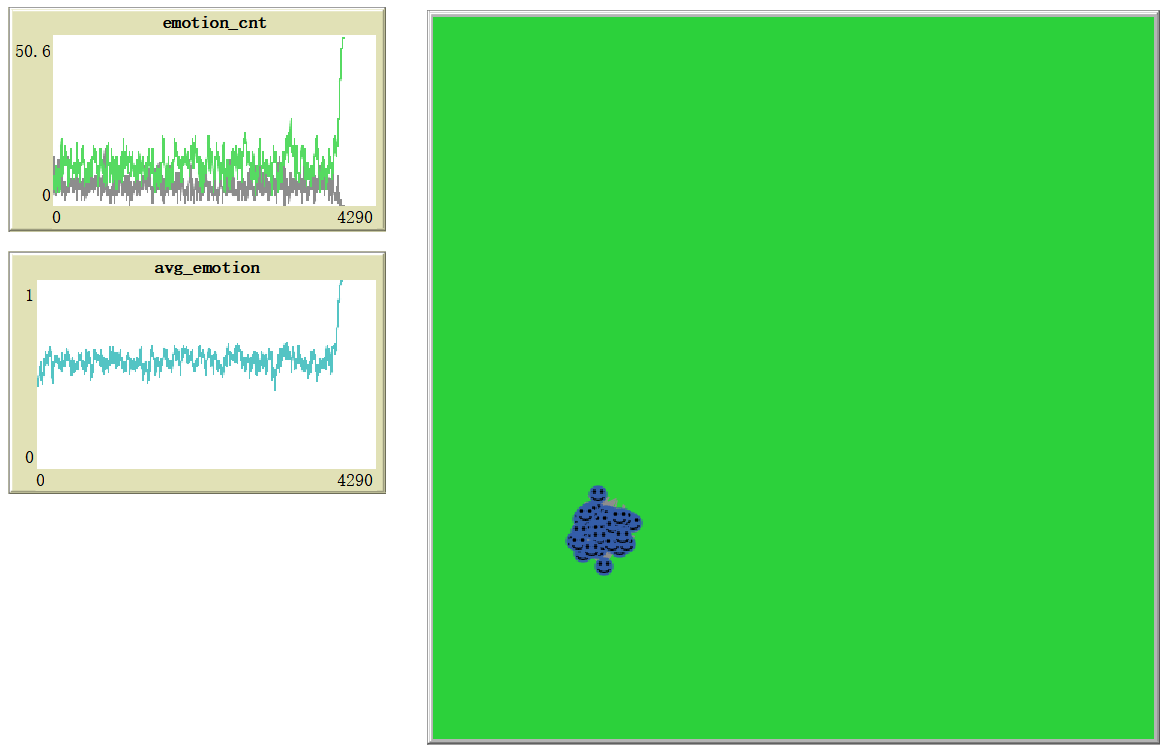


图4.2 坏消息传播半径是好消息3倍的实验结果

将坏消息的传播半径增加到好消息的4倍时，模型将无法收敛。用户的平均心情值在0.5左右波动。不过，交流环境也没有继续恶化，如图4.3所示。

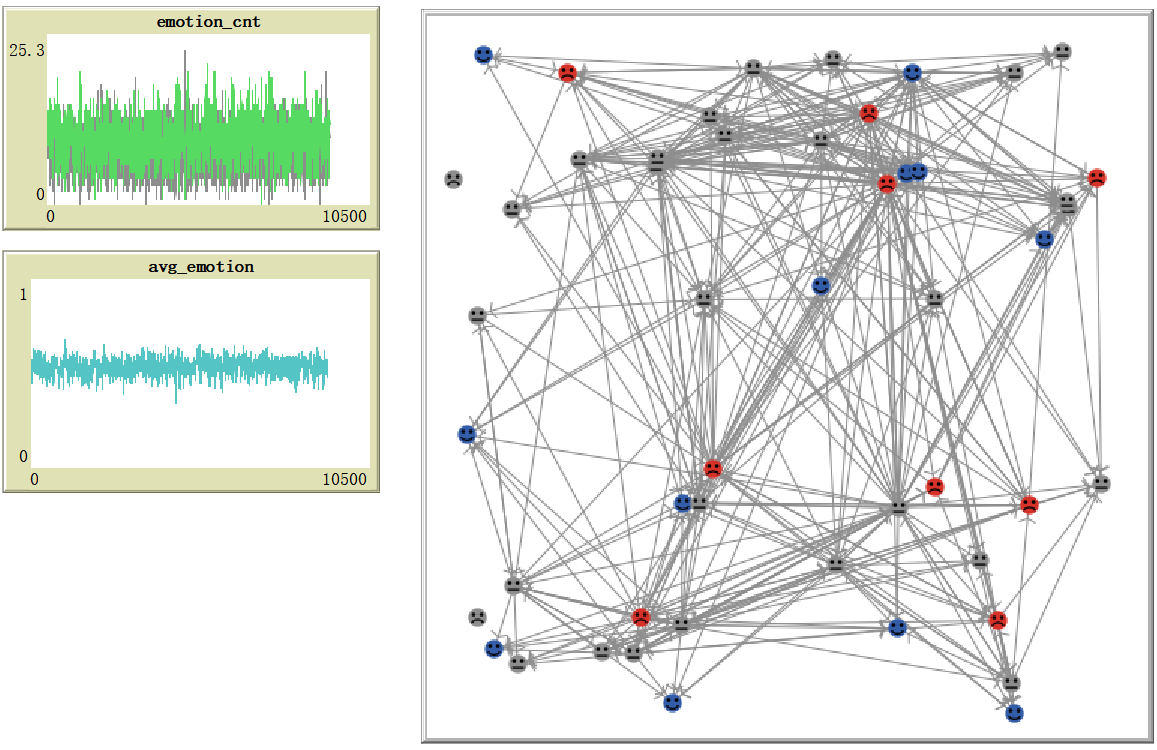


图4.3 坏消息传播半径是好消息4倍的实验结果

那么，更严厉的惩罚措施是否会让结果变好？本文将惩罚机制改为：只要发送过一次不良信息就封号。结果显示，模型仍然不会收敛，因为心情值很高的用户仍然有概率发送不良信息而被立刻封号，几乎不会有老用户留存下来，平均心情值仍然在0.5左右波动。可见，惩罚机制并非越严厉越好。影响力越大、发送不良信息越多的用户越要施以严厉的处罚；对于发表不良信息次数不那么高的用户，惩罚力度过重会导致大量用户流失，导致社区氛围难以得到持续的塑造。

**5 结论**

本文对社交网络进行了建模，根据模型设计了针对用户发言历史与影响力的惩罚机制，并评估了其在不同条件下的效果。实验结果表明，本文所设计的惩罚机制对于维护社交网络的交流环境质量有着良好的促进作用。

**参考文献**

# [1] QUEST MOBILE. QuestMobile2023年新媒体生态洞察：行业用户规模10.88亿，用户流转、分流进入新阶段，平台以两大途径谋增长、冲变现[EB/OL]. (2023-11-21)[2024-05-22].https://www.questmobile.com.cn/research/report/1726888249161519105