



中国研究生创新实践系列大赛
“华为杯”第二十一届中国研究生
数学建模竞赛

学 校 北京邮电大学

参赛队号 24100130040

队员姓名

1. 沈飞
2. 汪金锋
3. 张睿涵

中国研究生创新实践系列大赛

“华为杯”第二十一届中国研究生

数学建模竞赛

题 目： WLAN 组网中网络吞吐量建模

摘 要：

本文通过对 WLAN 网络中的接入点（AP）在不同网络环境下的行为进行建模和分析，旨在预测系统吞吐量并提出优化方案。研究内容分为三个主要部分，分别针对 WLAN 网络的信道接入机制、物理层速率（MCS 与 NSS）的选取、以及系统吞吐量的预测展开详细分析。

问题 1：信道接入分析

首先，本文探讨了 WLAN 网络中 AP 之间的信道接入竞争机制。由于多个 AP 和 STA（站点）在同一信道上通信，会产生竞争效应，影响网络的整体性能。通过对训练集中的数据分析，重点研究了 AP 发送数据帧序列的总时长、节点间 RSSI（接收信号强度指示）以及信道接入门限等信息，建立了 AP 发送数据包机会关于测试基本信息的模型，该模型刻画了信道接入的时机、信道占用时间，以及由于多 AP 干扰带来的信道访问失败问题。通过分析这些特征信息，可以初步预测 AP 在复杂信道环境中的信道接入行为，为吞吐量预测奠定基础。

问题 2：MCS 与 NSS 的选取

在 WLAN 网络中，AP 的数据传输速率通常由调制编码方案（MCS）和空间流数（NSS）决定。本文深入研究了如何根据网络中的物理层条件，如节点间的 RSSI 和信道质量，预测 AP 在数据传输中选用的 MCS 和 NSS 组合。通过对实测训练集的分析，AP 在传输过程中会选择一种最优的 MCS 和 NSS 配置，该配置在 AMC（自适应调制编码）算法的收敛后会保持较长时间。基于此假设，本文构建了 MCS 与 NSS 的预测模型，结合 RSSI 门限信息来决定 AP 的速率配置。虽然预测模型的精度存在一定的局限性，但通过数据中的统计信息可以获得较为稳定的速率配置。

问题 3：系统吞吐量的预测

最后，基于前两问的分析结果，包括帧时长、MCS、NSS 等特征，利用多层感知机 (Multilayer Perceptron, MLP) 预测了为 WLAN 网络的系统吞吐量预测模型引入的额外特征。模型的输入特征涵盖了物理层速率、信道占用时间、信道条件和干扰情况等。最后本文使

用回归模型和集成学习方法对吞吐量进行预测，并通过交叉验证与超参数调优提升预测精度。研究结果可为 WLAN 网络的优化提供指导，通过改善 MCS 和 NSS 的选择、改善信道环境、减少 AP 间干扰等手段，有望提升网络吞吐量。在前两问的基础上，本文进一步研究了系统吞吐量的预测问题。系统吞吐量是 WLAN 网络性能的关键指标，受多个因素影响，如物理层速率、信道接入时长、干扰等。基于问题一中分析的信道接入时长，问题二中预测的 MCS 与 NSS 组合，以及其他基础网络测试信息，本文构建了 WLAN 系统吞吐量的回归预测模型。模型的输入特征包括：AP 发送的总时长预测得到的 MCS 与 NSS 组合节点间 RSSI 及其门限信道占用情况和干扰因素

在模型的构建过程中，本文采用了回归模型和集成学习方法（如随机森林，XGBoost 等），本文指出，虽然 MCS 与 NSS 的预测精度较低，但仍可以通过实际统计的真实数值对模型进行调整，保证系统吞吐量预测的精确性。

本文提出的吞吐量预测模型不仅为 WLAN 网络性能分析提供了有效的工具，还为未来的网络优化提出了切实可行的方案。通过对信道竞争机制的深入理解，可以优化 AP 的信道接入策略，减少多 AP 之间的冲突和干扰，提升整体网络的吞吐量。此外，本文的研究成果可应用于工业、教育、医疗等多种场景，通过优化 WLAN 网络配置，提高用户的通信质量和业务体验。

总结而言，本文通过信道接入、MCS 与 NSS 选取，以及吞吐量预测的系统性分析，成功构建了 WLAN 网络的性能预测框架，为未来的无线网络优化提供了理论和实践上的指导。

关键词： WLAN 系统 吞吐量 RSSI SINR

1 问题重述

1.1 问题背景

无线局域网（Wireless Local Area Network, WLAN）以人们的业务诉求为驱动力，部署场景日益增多，在无线化办公、教育、医疗、工业制造、仓储等场景都有广泛应用。尽管下一代 Wi-Fi7 标准的峰值速率达到 30Gbps，但是在例如小区、商场、学校等高密部署场景下，相邻节点之间覆盖范围重叠，信号干扰碰撞问题突出，使得部署带宽和数据传输速率大幅下降。为了进一步提升系统吞吐量，对 WLAN 系统的进一步优化具有重要意义。

精准和快速的吞吐量预测是 WLAN 优化问题的基础问题，能够大幅度提升 WLAN 系统的鲁棒性和性能（吞吐量指节点单位时间内成功发送的比特数）。当前的一些研究通过机器学习方法，提取各节点之间的基本信息、架构信息以及节点动态位置、动态干扰等临时信息，进行训练和建模。但是由于研究中使用的仿真方法并不能精确反应实际部署中快速变化的信道环境、干扰以及复杂多样的业务流，其结果在真正商用中并不能达到预期实用价值。

因此，为了对 WLAN 进行优化，在工业、教育、医疗等新场景中，提升用户体验，设计一种能够基于 WLAN 实测数据，分析 WLAN 网络拓扑、节点间 RSSI、信道接入机制、干扰等因素对 WLAN 数据发送、速率的影响，进一步地，对 WLAN 系统吞吐量进行精确预测的模型具有重要研究意义。

1.2 问题描述

基于上述研究背景，题目提供了 WLAN 组网场景下的实测数据，数据集提供了包括测试基本信息，以及测试中收集的数据帧相关统计信息，围绕 WLAN 组网中网络吞吐量建模，提出以下问题：

问题 1：根据附件 WLAN 网络实测训练集中所提供的网络拓扑、业务流量、门限、节点间 RSSI 的测试基本信息，分析其中各参数对 AP 发送机会的影响，并给出影响性强弱的顺序。通过训练的模型，预测每个 AP 的发送机会，即发送数据帧序列的总时长（seq_time），并通过测试集 test_set_1_2ap 和 test_set_1_3ap（仅提供模型输入信息）预测 AP 发送数据帧序列的总时长。可按照同频 AP 个数分类分析和分别建模，也可统一分析和建模。

问题 2：根据附件提供的实测训练集中的测试基本信息，特别是节点间 RSSI 信息和门限信息，结合问题 1 中对 AP 发送机会的分析，对测试中 AP 发送数据选用最多次数的（MCS, NSS）进行建模，并通过测试集（仅提供模型输入信息）预测（MCS, NSS）。

问题 3：结合问题 1 和问题 2 的分析，对系统吞吐量进行建模，并通过测试集 test_set_1_2ap 和 test_set_1_3ap 预测网络吞吐量。可按照同频 AP 个数分类建模，也可统一建模。本问题允许采用实测中统计的数据帧真实（MCS, NSS）作为模型输入变量。

2 模型假设与符号说明

2.1 模型假设

1. 假设任意两 AP 之间在互相影响的效力上具有对等地位，对各自有同样的干扰；
2. 假设基站位置信息对 AP 之间干扰的影响，仅通过 RSSI 值来体现；
3. 假设在传输方式为同步的时候，环境底噪为-100dbm；

2.2 符号说明

符号	意义
T_{seq_time}	数据帧发送总时长
T_{PPDU}	发送一个数据包的平均时长
L_{PPDU}	一个 PPDU 数据包的长度
L_{MPDU}	一个 MPDU 数据包的长度
L_{MSDU}	一个 MSDU 数据包的长度
N_{AMPDU}	一次 AMPDU 聚合过程中使用的 MPDU 数据包个数
N_{AMSDU}	一次 AMSDU 聚合过程中使用的 MSDU 数据包个数
PER	丢包率
Throughput	吞吐量
Trans	传输方式

3 基本原理

3.1 接入机制

WLAN 中工作在同一信道的各节点共享信道，节点通过载波侦听多址接入/退避的机制避免冲突。接入过程可分为以下 3 个步骤：

- (1) 信道可用评估 (Clear Channel Assessment, CCA)：节点有数据要发送时，首先对工作信道进行固定时长的载波侦听，这个固定时长被称为分布式协调帧间距 (Distributed Coordination Function Inter-frame Space, DIFS)。如果 DIFS 时段内接收到的信号强度 RSSI 低于 CCA 门限，判断信道为空闲，否则判断信道为繁忙。
- (2) 随机回退：判断信道为空闲时，为避免节点间碰撞，每个节点根据其竞争窗口 (Contention Window, CW)，从 $[0, CW-1]$ 的整数均匀分布选取一个随机整数作为回退数，将其乘以时隙 slotTime ($9\mu s$)，称为随机回退时段。如果信道在随机回退时段保持空闲，则节点开始一次数据传输。如果期间信道变繁忙，节点将暂停回退，直到信道重新在 DIFS 内空闲，再继续前面的回退。随机回退采用二进制指数退避算法确定。
- (3) 数据传输：回退到 0 的节点发送一个数据帧，接收节点在成功接收到数据之后等待短帧帧间距 (Short Inter-frame Space, SIFS) $16\mu s$ ，回复确认帧 (Acknowledgement, ACK) $32\mu s$ 。

3.2 CCA 门限

WLAN 有两层协议栈，即物理 (Physical, PHY) 层和媒体接入控制 (Medium Access Control, MAC) 层。数据字段首先在 MAC 层封装 MAC 头，携带源 MAC 地址和目的 MAC 地址，接着在 PHY 层封装上 PHY 头，用于调制和解码，形成一个完整的 Wi-Fi 帧被发送出去。信道中传输的数据可以被一定区域内的任何其他节点接收，节点首先对 PHY 头解码，将解封装的数据帧上送到 MAC 层，节点通过 MAC 头携带的目的地址识别发送给自己的数据。AP 通常配 8 根不同角度的天线接收信号，所有天线接收 RSSI 的和用于解码。WLAN 引入包检测 (Packet Detection, PD) 门限和能量检测 (Energy Detection, ED) 门限这两个 CCA 门限，典型值分别是 -82dBm 和 -62dBm 。若节点 PHY 层接收机检测到完整的数据包的 PHY 头的前导 (Preamble)，则判定为 Wi-Fi 报文，采用 PD 门限；若未检测到 Preamble，判定为非 Wi-Fi 报文，则采用 ED 门限判断信道忙闲。AP 通常配 8 根不同角度的天线接收信号，各天线所接收 RSSI 的和就是所接收信号的 RSSI。然而，在进行 PD 和 ED 门限判决时，仅采用各天线所接收 RSSI 中的最大值。

3.3 NAV 机制

为保障低延时等高优先级业务，WLAN 支持在 MAC 头携带网络分配矢量 (Network Allocator Vector, NAV) 字段，指示其它节点在该 NAV 时段内静默来为自己清空信道。NAV

时段为该节点从发送数据帧到接收完 ACK 的时长。一个节点若处于其它节点指示的 NAV 静默期，则不竞争信道。在组网中，当 AP PHY 层接收到来自相邻 BSS 的数据帧时，若 RSSI 低于 NAV 门限（通常为-82dBm），则停止接收，直接丢弃；若 RSSI 高于 NAV 门限，则完成接收，并将其上送到 MAC 层，更新 MAC 头中携带的 NAV 时段。关于信道忙闲按 CCA 门限判断并结合静默期决定。AP 采用各天线所接收 RSSI 的平均值用于 NAV 门限判决。

3.4 随机回退

随机回退采用二进制指数退避算法确定回退时间。CW 的初始值为 CW_{min} ，不同节点的 CW_{min} 可以相同，也可以不同，每次数据传输失败后进行重传时，CW 翻倍。如果 CW 达到了 CW_{max} ，则保持此值，直到被重置为止。每次数据传输成功时 CW 重置，开始下一个数据帧的回退。若传输连续失败，重传次数达到 r 后，数据帧被丢弃，CW 重置传输下一个数据帧。可见，重传 r 次时，无论成功还是失败，CW 都会重置。

附图3.1中的一次传输（Tx, transmission）包含了发送一个数据包和接收一个 ACK，一次 collision 包含了发送一个数据包和等待 ACKTimeout 时长。帧序列如附图 2 所示，一个数据帧包括 PHY 头、MAC 头和有效载荷 payload，ACK 时长为 $32\mu s$ ，ACKTimeout 为 $53\mu s$ 。为减少冲突，AP 采用 RTS-CTS 模式发送数据，即发送节点先发送一个请求发送（Request to Send, RTS）帧，收到的节点回复一个允许发送（Clear to Send, CTS）帧，RTS 和 CTS 的 MAC 头中携带 NAV，通过该方式通知发送节点和接收节点的通信范围内的其他节点在此期间静默。RTS 和 CTS 是短帧，时长 $32\mu s$ ，RTS 或 CTS 碰撞，相比数据帧碰撞，可避免信道浪费。RTS-CTS 模式下的帧序列如图3.2所示。

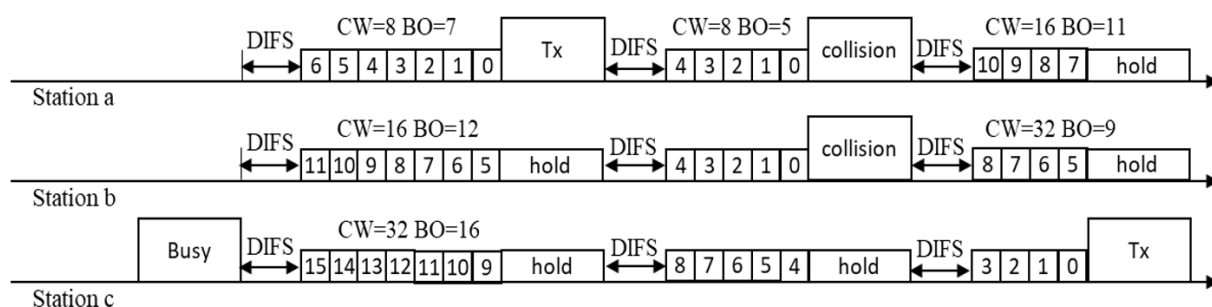


图 3.1 二进制指数退避过程

3.5 聚合

为了提升发送小包的效率，协议允许通过聚合一次发送多个具有相同目的地址的数据包。来自网络上层（如以太网逻辑链路层 LLC 层），具有相同接收地址的同服务类别的 MAC 服务数据单元（MAC Service Data Unit, MSDU），可以在 MAC 层顶端被拼接起来，加

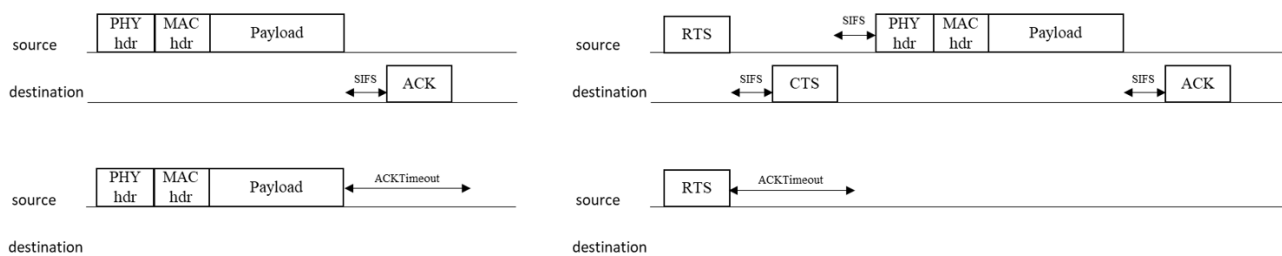


图 3.2 数据帧序列和 RTS-CTS 模式下发送成功发送失败的帧序列

上一个共同的 MAC 头，封装成一个 MAC 协议数据单元（MAC Protocol Service Data Unit, MPDU），这个过程叫作聚合 MSDU（Aggregated MPDU, AMSDU）。组装好的 MPDU 在 MAC 层底端被聚合起来，每个 MPDU 前加一个短分隔符，随后聚合成为一个 PHY 协议数据单元（PHY Protocol Data Unit, PPDU）送给 PHY 层。每次发送数据时，缓存里有包则聚，遵循 N_{AMSDU} 个数最多为 7， N_{AMPDU} 个数最多为 21，同时，一个 PPDU 时长不超过 4.5ms。

在进行聚合时，由于聚合的子 MSDU 之间，及子 MPDU 之间需要插入冗余字段，实际一个 PPDU 传输时长里，仅 MSDU 的总字节数是有效传输数据，进行吞吐量计算时才被计入，因此，AMS DU 和 AMPDU 聚合的准确评估对吞吐量预测也有影响。如附图 3（a）所示，应用层报文 payload 长度为 1420 Bytes，包含 40 Bytes 长的 IP 协议头和 1380 Bytes 长的有效载荷。经过以太网有线网到达 Wi-Fi 的 MAC 层后，不聚合，加上 8 Bytes 的 LLC 层（以太网逻辑链路层）头，封装上 30 Bytes 的 MAC 头（802.11 header），形成一个 MPDU。如 3.3 所示，缓存里包增多时，进行聚合，MSDU 之间插入 0-3 字节的填充字节进行 4 字节对齐，一个 MPDU 中的 MSDU 共用一个 30 Bytes 的 MAC 头。MPDU 进行聚合时，前面加上分隔符 delimiter，再进行 4 字节对齐，一个 PPDU 封装一个 PHY 头。

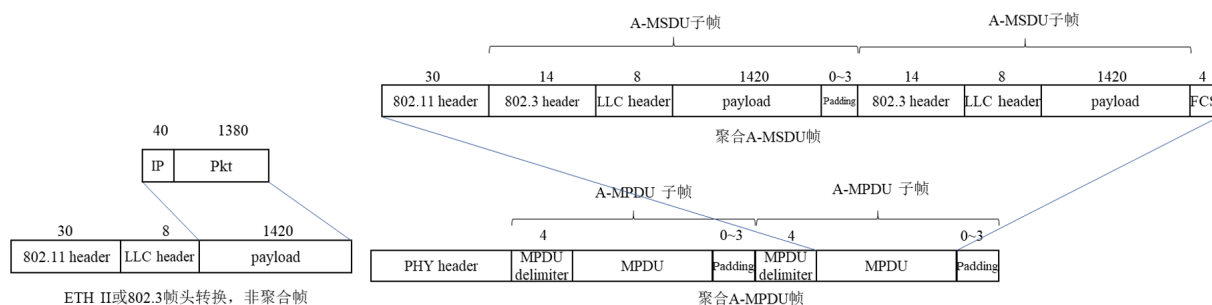


图 3.3 （a）以太网帧不聚合（b）AMSDU 聚合和 AMPDU 聚合

3.6 传输方式

当两个节点间的 $\text{RSSI} > \text{ED}$ 时，一个节点传输数据时，另一节点检测信道为繁忙，称两个节点能“听”到。那么多数情况下二者的数据传输将交替进行，只在偶然同时回退到 0

时同时开始数据传输。节点同时发送数据时产生碰撞，可能导致传输失败。这种在节点间可侦听情况下发生的传输称为同步传输，包括交替进行和同时发生的传输。

WLAN 实际部署中，受覆盖范围和可用信道数约束，工作在相同信道的同频 AP 间 RSSI 大多处于 [PD, ED] 区间，一个区域内同频 AP 数量通常为 3 5 个。受业务类型影响，包长差异较大，如3.4所示，在某次同步传输过程中，先结束传输的 AP2 进行 CCA 时，由于已经错过侦听 AP1 的 Preamble，将采用 ED 作为 CCA 门限，从而判定信道为空闲，在回退到 0 时开始一次新的传输。称为异步传输。若 AP1 结束传输时，AP2 的第二次传输已开始，则 AP1 同样原因可能会开始第二次传输，那么两个 AP 可能进入长时间的异步传输状态。

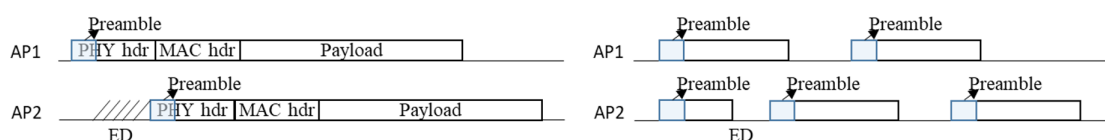


图 3.4 一次异步传输和异步传输状态示意图

3.7 自适应调制编码算法

发送数据的 PHY 层速率 (PHY Rate) 由调制编码方案 (Modulation and Coding Scheme, MCS) 和空间流数 (Number of Spatial Stream, NSS) 表征，一组 (MCS, NSS) 对应一个 PHY Rate (见 4.2 节)。MCS 和 NSS 越高，发送时携带的有效比特数越多，即 PHY Rate 越高。节点的 PHY 层对信号进行解调时，要求一定的 SINR，其 SINR 越高，可支持成功解调的 MCS 和 NSS 越高。WLAN 采用经典的自适应调制解调 (Adaptive Modulation and Coding, AMC) 算法，根据信道条件动态调整发送数据使用的 (MCS, NSS)。具体地，初始化时采用默认值 (如 MCS6, NSS2) 发送数据，并持续统计和更新丢包率 (Packet Error Rate, PER)。当 PER 高于一定阈值时，代表当前 SINR 下，以该 (MCS, NSS) 发送数据的解调成功率低，需要降低 MCS 和 NSS；否则提高 MCS 和 NSS。AMC 算法自适应寻找最优的 PHY Rate 发送数据，使得吞吐量最大。

4 问题一的分析与求解

4.1 问题分析

4.1.1 问题简述

问题一主要研究 AP 发送的机会，旨在探索 WLAN 网络中的节点在测试场景下发送数据包的机会，即发送数据帧列的总时长 (seq_time)，并构建预测模型，该问题包括两个子问题：

(1) 分析测试基本信息中各参数对 seq_time 的影响并排序：

首先，基于训练集中 $\text{training_set_2ap_loc0_nav82}$ 等 13 个数据表，对同频 AP 个数为 2 和同频 AP 个数为 3 的场景分别进行建模分析。对于每一种同频 AP 类型，研究表中所提供的网络拓扑、业务流量、门限、节点间 RSSI 的测试基本信息，分析其中各参数对 AP 发送机会的影响，即对 seq_time 数据值的影响，并根据其影响性强弱给出重要性排序。

(2) 构建模型预测每个 AP 的 seq_time ：

根据训练集中 $\text{training_set_2ap_loc0_nav82}$ 等 13 个数据表，分别对同频 AP 个数为 2 和同频 AP 个数为 3 构建模型预测测试集 test_set_1_2ap 和 test_set_1_3ap 中 AP 发送数据帧序列的总时长，然后将模型应用于测试集，给出测试集中 seq_time 的预测值。

通过解决问题一的两个子问题，我们将更好地了解哪些因素能够影响 AP 发送数据的机会，并能够进一步研究各参数对 WLAN 网络吞吐量的影响。这对于无线局域网系统的进一步优化具有重要意义。

4.1.2 机制分析

根据题目，在节点间 RSSI、信道竞争接入机制、CCA 门限、NAV 机制等共同影响下，节点以一定的概率发送数据。故想要准确预测 AP 发送数据的机会，需要分别考虑上述各因素影响与综合影响，本节具体分析论述了其影响机制与运行方式。

(1) 传输方式

节点之间的接收信号强度 (Received Signal Strength Indication, RSSI) 不仅体现了节点间的位置信息，还将与 CCA、NAV 机制共同决定节点能否争夺到信道和发出数据包。根据两节点能否互“听”，以及 NAV 机制是否有效，节点间的传输方式分为了同步传输、异步传输以及同步异步混合传输。

1、同步传输：此时 $\text{AP 间 RSSI_MAX} > \text{PD}$ 且 $\text{RSSI_MEAN} > \text{NAV}$ ，两 AP 之间互听，交替抢到信道，偶然同时发送，且 NAV 生效。STA 接受数据时干扰主要来自环境底噪。

2、异步传输：此时 $\text{AP 间 RSSI_MAX} < \text{PD}$ 且 $\text{RSSI_MEAN} < \text{NAV}$ ，两 AP 之间不互听，邻区干扰小，同时进行数据发送，NAV 不生效。此时 STA 接收数据时可能受到较小的邻区干扰和环境底噪。

3、宏观上同步与异步传输混合：此时两个 AP 的 $RSSI_MAX > PD$ 且 $RSSI_MEAN < NAV$ ，两 AP 之间互听，邻区干扰大，NAV 不生效。当传输中出现由于错过 Preamble 时，无法利用 NAV 更新通知其他 AP，导致异步传输。此时 STA 接收数据时受到的干扰根据具体处于的传输方式决定。

同步传输方式的优点是各 AP 在预定时隙内传输，避免了信道竞争和冲突，保证了公平性和确定的传输时机，并减少了干扰，即使 SINR 较低的 AP 也可以获得稳定的传输时机。但其缺点在于固定时隙可能造成资源浪费，且对负载变化的灵活性较差，导致潜在的吞吐量瓶颈。如果负载不均衡，某些 AP 可能会浪费时隙资源。

异步传输方式的优点是灵活性高，AP 可以随时尝试传输，动态利用信道资源，适应负载变化。当信道负载较低时，异步传输方式更有效率。然而，其缺点是信道竞争可能导致碰撞、延迟和不公平分配，对于 SINR 较低的 AP，可能因竞争失败而传输机会减少。尤其在高密度环境中容易出现干扰和“捕获效应”，即较强的 AP 总是比较弱的 AP 占据更多的信道资源。

(2) 信干噪比 (SINR)

SINR 是衡量信号质量的重要指标，其值越高，表示信号质量越好，节点间的传输效果越好。因此，我们认为节点所处环境的 SINR 值对于节点发送数据的机会有着重要的影响。

第 (1) 节中所分出的同步和异步**传输方式**，导致 AP 能收到不同类型的干扰，使得计算 SINR 值的方式不同。在同步传输情况下，干扰仅包括环境底噪，而在异步传输情况下，STA 接收数据时还可能受到邻区 AP 的干扰。除了传输方式外，我们发现，不同的**同频 AP 个数**也会导致 SINR 值的计算方式不同。在同频 AP 个数为 2 的场景下，AP 仅需要考虑另一相邻 AP 的干扰，而在同频 AP 个数为 3 的场景下，各 AP 均需综合考虑另外两个 AP 的干扰。特别是在异步传输场景下，由于 STA 的传输受到邻区干扰，这一影响尤为明显。为此，我们调研了如何在 3AP 场景下叠加两个不同干扰源的方法，以表征其对信干噪比 SINR 的影响。

据调研 [1]，接收信号强度指示 (RSSI) 可以通过多种方式计算，常见的计算方式包括线性功率值和对数功率值。不同的方式可以结合叠加使用来获得不同情境下的总接收信号强度。

1. 线性功率值

RSSI 可以直接用功率值来表示，通常以瓦特 (W) 或毫瓦 (mW) 为单位。多个信号的线性功率值可以通过直接加和来叠加：

$$P_{\text{total}} = P_1 + P_2 + \cdots + P_n$$

其中 P_i 表示每个信号的功率值。

2. 对数功率值

在对数尺度下，如以分贝（dB）为单位表示 RSSI，信号强度的叠加需要将对数值转换回线性功率值，再进行加和：

$$P_{\text{total}} = 10 \log_{10} (10^{P_1/10} + 10^{P_2/10} + \dots + 10^{P_n/10})$$

考虑到 RSSI 具有上述两种不同的叠加方法，在计算同频 AP 个数为 3 的场景中的邻区干扰时，我们使用对数方法进行信号强度叠加，随后使用线性方法计算关联 AP 到该 STA 的 RSSI 与邻区 AP 到该 STA 的 RSSI 之差。另外，通过简单观察训练集中各接收到 RSSI 参数的取值大小，发现其在测试中出现的可能范围于 -98dBm 至 -29dBm 之间，我们规定 $RSSI_{\text{noise}} = -100\text{dBm}$ 为环境噪声强度。

综上，结合传输方式和同频 AP 个数两个因素，我们使用不同方法对 SINR 进行计算，使用 ‘+’, ‘-’ 号表示线性叠加，使用 \oplus 号表示对数叠加。具体计算公式如表 4.1：

同频 AP 个数	同步	异步
2	$RSSI_{\text{self_ap}} - RSSI_{\text{noise}}$	$RSSI_{\text{self_ap}} - RSSI_{\text{other_ap}}$
3	$RSSI_{\text{self_ap}} - RSSI_{\text{noise}}$	$RSSI_{\text{self_ap}} - (RSSI_{\text{other_ap1}} \oplus RSSI_{\text{other_ap2}})$

表 4.1 STA 的 SINR 计算方法

其中， $RSSI_{\text{self_ap}}$ 表示 STA 接收到的自身 AP 的信号强度， $RSSI_{\text{other_ap}}$ 表示 STA 接收到的邻区 AP 的信号强度， $RSSI_{\text{noise}}$ 表示环境噪声强度。

(3) 总结：

a) RSSI 作为题目给定的重要特征，体现了节点接收到来自其他拓扑中节点的信号强度。经过分析，该特征由于经过去底噪处理，时序特征较弱，且在测试期间波动不明显。因此，我们在本题中考虑将 RSSI 序列用于判断节点接受的信号强度所处 CAA 和 NAV 门限区间，随后，对 RSSI 序列求平均，用单值替代，作为节点间的定值特征。

b) 当节点接受到的 RSSI 较小时，即其他节点信号影响较小时，AP 根据对信道忙闲的判断更大可能性为空闲，并发送数据。尽管此时仍有随即回退机制的限制，但可以推断的是，此时 AP 发送数据的机会更大，并导致更长的 seq_time。相反，当节点接受到的 RSSI 较大时，其他节点信号影响较大，此时 CCA 机制和 NAV 机制生效的概率更高，节点需要遵守接入机制，在预定时隙内传输，与相邻节点共用信道资源，具有有限的发送数据的机会，即 seq_time 较小。因此，我们可以根据对 AP 所听到的 RSSI 强弱，判断其所处的门限区间，以进一步根据其传输方式进行分类，用于本题模型预测。

c) 信干噪比（SINR）在无线通信中对 AP（接入点）发送机会的影响，可以理解为 SINR 决定了每个 AP 在竞争信道时的胜率以及发送数据帧的效率。具体来说，SINR 越高，

意味着信号强度相对于干扰和噪声的比例越大，数据传输的成功率越高，信道的利用率也会相应增加。然而，在 WLAN 网络环境中，环境噪声和邻区干扰等因素都会对 SINR 产生影响，随着拓扑中同频 AP 个数的增加，邻区干扰进一步加剧了这种影响。针对不同因素环境，采用不同的 SINR 的计算方式，能够更精确地模拟测试中节点间信号干扰情况，进而更好地预测 AP 发送数据的机会。

综上所述，我们认为节点的 RSSI 信息、传输方式、节点 SINR 值以及门限值对预测 AP 的发送机会都具有一定影响。对 AP 发送机会的预测可表示为：

$$T_{seq_time} = f(RSSI_*, Trans, SINR_{sta}, pd, ed, nav)$$

其中， $RSSI_*$ 表示各类 RSSI 信息， $Trans$ 表示传输方式， $SINR_{sta}$ 表示信干噪比， pd, ed, nav 表示门限信息。

4.1.3 求解思路

针对上述问题，提出如下求解思路：

由题目可知，WLAN 部署后，节点基于信道竞争接入机制进行 CCA 和随机回退，并发送数据。在节点间 RSSI、信道竞争接入机制、CCA 门限、NAV 机制等共同影响下，节点以一定的概率发送数据。因此，节点在某一功率下的某一时刻听到的 **RSSI 处于什么范围（是否超过门限）**，会使得节点因此选择传输、等待（DIFS/回退）或静默，乃至导致可能的测试组内的传输方式（同步/异步），这一机制能够很大程度上影响对节点数据帧发送时长的预测，因此除了使用训练集给定的参数外，我们还需要从这些给定数据中挖掘对应新特征表示节点状态以更好地进行预测。问题一的求解思路如下图4.5：

4.2 问题求解

4.2.1 数据处理

针对所提供的网络拓扑、业务流量、门限、节点间 RSSI 的测试基本信息包括共 35 个可选特征变量，我们进行预处理，进行以下步骤操作。

（1）数据清洗，我们将相同同频 AP 个数的所有训练表合并为一个主表，对于表中观测到的异常空值，和异常数值（例如在 `training_set_3ap_loc30_nav86` 中部分列数据为空，以及 `training_set_2ap_loc2_nav82` 最后两行的数据与其他数据相比明显异常，数据与表头内容对不上），对此处理为删除整行数据避免对分析结果的影响。

（2）方差过滤，进行部分特征变量的移除，这些变量部分是常量（即在所有数据中保持不变），部分可以以行数奇偶性为隐藏属性表示，部分只是固有属性根据经验判断与因变量 `seq_time` 无关。删除变量包括：`test_dur`、`loc_id`、`pkt_len`、`bss_id`、`ap_name`、`ap_mac`、`ap_id`、`pd`、`ed`、`sta_mac`、`sta_id` 共 11 个变量。

（3）特征创造。a. 题目中提到 AP 传输方式对 AP 发送时长具有直接影响，考虑到 `pd`，

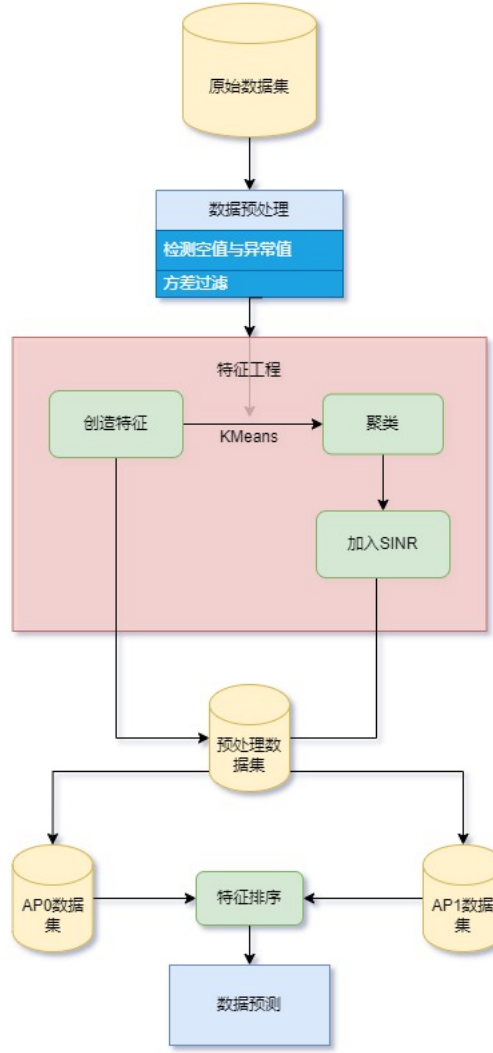


图 4.5 问题一思路示意图

ed 表现为信道传输门限，它们决定了 AP 的 CCA 机制，进而导致不同的可能传输机制，因此我们采用差值变量，以概率形式表示在某一时间点 AP 的传输方式可能性。变换公式如下：

$$pro_pd = \frac{num_{from_ap_max_rssi_gt_pd}}{num_{from_ap_rssi}} \quad (4.1)$$

$$pro_ed = \frac{num_{from_ap_max_rssi_gt_ed}}{num_{from_ap_rssi}} \quad (4.2)$$

$$pro_nav = \frac{num_{from_ap_mean_rssi_gt_nav}}{num_{from_ap_rssi}} \quad (4.3)$$

$num_{from_ap_max_rssi_gt_pd}$ 表示在某一时间点 AP 听到的 RSSI 大于 PD 门限的次数, $num_{from_ap_rssi}$

表示在测试期间 AP 听到的 RSSI 的总次数。同理可得 pro_ed , pro_nav 。b. 还引入了 SINR, 考虑了 STA 接收数据时可能受到的干扰。

(4) 聚类分析。根据 (3) 中创造的三个特征变量 (在 3AP 情况是两组共 6 个): pro_pd , pro_ed , pro_nav , 分别表示了某 AP 在一次测试期间接收到来自其他组内 AP 的 RSSI 高于 PD、ED、NAV 门限的频率, 这一组数据体现了此次传输中各测量时刻该 AP 所接受到的 RSSI 数值所分布范围, 以此近似表示该 AP 进行对应接入操作的概率。由于新特征呈现连续性, 取值处于 $[0, 1]$ 之间, 且具有不均匀性, 为了在进行门限区间分类时使用准确的阈值, 我们采用 K-means 聚类辅助分析:

K-means 是一种基于划分的无监督学习算法, 旨在将数据点划分为 K 个簇 (cluster), 使得簇内数据点的相似度尽可能高, 而不同簇之间的差异尽可能大。其核心思想是通过迭代优化, 将数据点分配给最近的簇中心 (centroid)。K-means 适用于聚类分析, 具有效率高、快速收敛的特点, 适合快速给出可视化聚类效果, 辅助进行本题的 RSSI 状态分析。

通过肘部图以及轮廓系数共同推断 (图4.6), 分类分成三簇, 符合上文机制分析中对 AP 传输方式的推断。

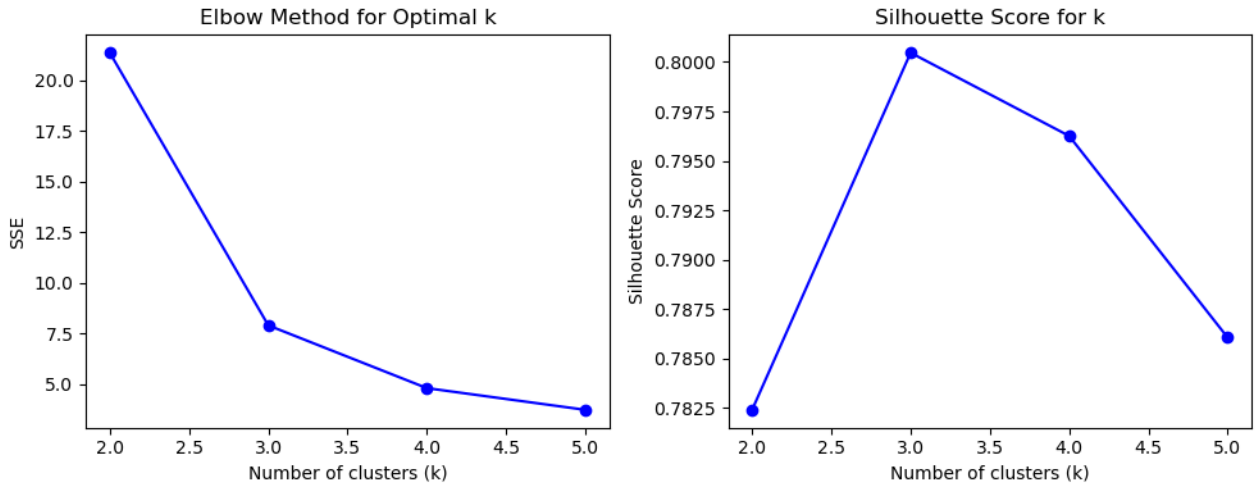


图 4.6 肘部图 (左) 以及轮廓系数 (右)

聚类结果如下图4.7, 表4.2。

创建新的特征变量'SINR', 根据分类类型, 使用 5.1.2 (2) 节提到的计算方法分别计算对应 SINR 值。

(5) 数据变换。对 protocol 变量, 取 01 变换。筛选并去除 RSSI 异常值, 构造 pro_pd 等之前没有筛选, 这是为了保证得到的概率值更具真实性, 现在使用 3σ 准则筛选, 去除异常值。随后, 为了得到更好的预测效果, 对这些数列求平均, 得到各 RSSI 序列的平均强度。

(6) 主表拆分,观测到训练数据中, 每个测试由 AP 个数行组成, 考虑到这些行之间

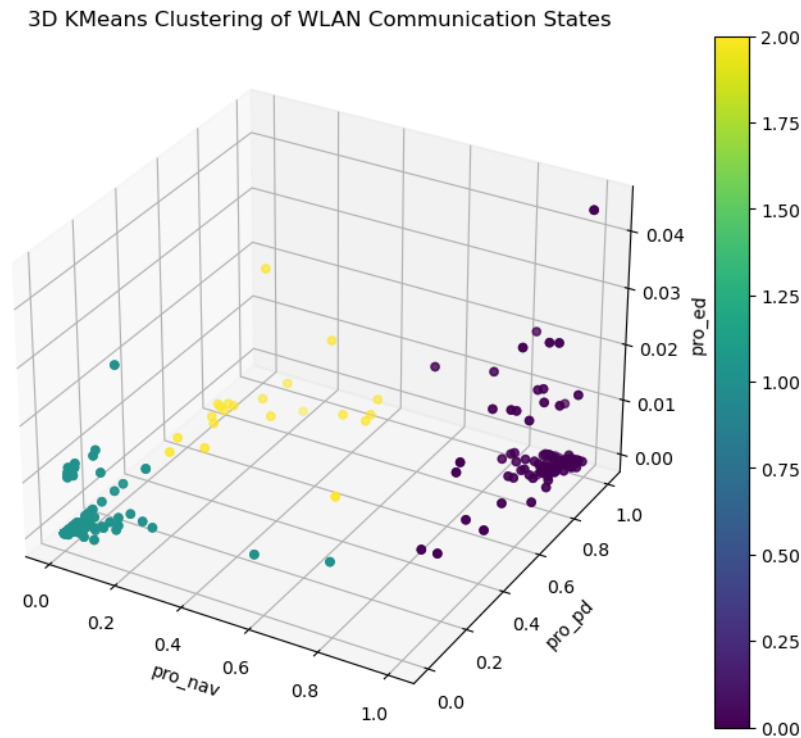


图 4.7 K-eans 聚类结果

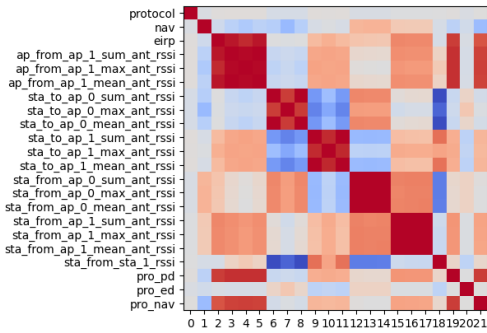
	pro_pd	pro_ed	pro_nav
0	0.897140	0.003123	0.924633
1	0.063595	0.001679	0.034990
2	0.770176	0.001579	0.150017

表 4.2 聚类所对应门限区间概率

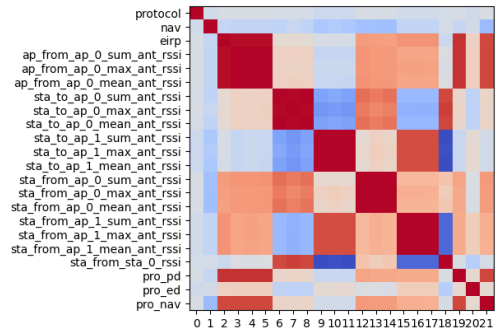
由于在同一个测试场景中，部分 RSSI 数据具有互补性，将其分割为 AP 个数个子表，对于 AP=2，按照行数对 2 的余数分为两个子表，对于 AP=3，按照行数对 3 的余数分为 3 个子表。每个子表对应一个 AP 在测试中的表现。此时观察到 test_id 在子表无意义，移除该变量。

(7) 相关性分析，接着我们对特征变量: 协议, RSSI, nav, pro_pd, pro_ed 等进行相关性分析（图4.8）再次剔除了一些具有强线性相关的特征变量。其中在此基础上，erip 与 ap_from_ap_other_sum/max/mean_ant_rssi 具有强正向线性关系，将其结合在一起考虑。另外，sta_to_ap_other_sum/max/mean_ant_rssi 和 sta_from_ap_other_sum/max_mean_ant_rssi 等也汇总在一起考虑，使用 sum_rssi 来代表这些 max/mean rssi 指标。因为我们认为 sum 值

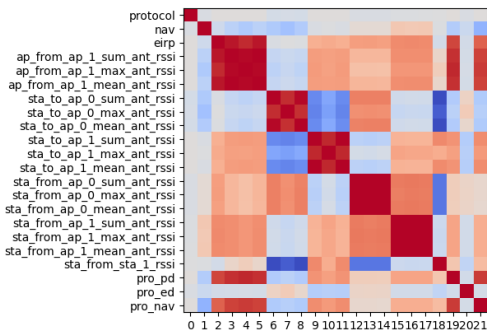
相较于前两者有更多信息，更具有代表性。



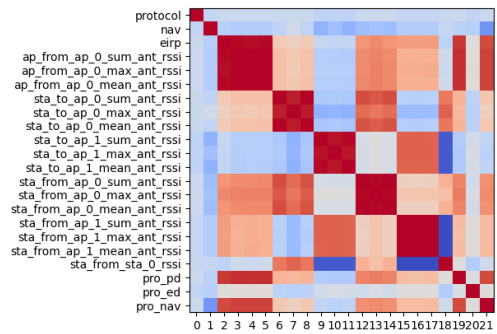
(a) AP0 的 Pearson 相关系数



(b) AP1 的 Pearson 相关系数



(c) AP0 的 spearman 相关系数



(d) AP1 的 spearman 相关系数

图 4.8 AP0 和 AP1 的相关性分析

(8) 子表合并，为了提升预测模型输入数据集的大小，我们将 AP=2 和 AP=3 的子表分别合并，形成一个新的数据集，以便于后续建模。合并过程中，对主表中互补的特征变量进行对应合并，例如，在 AP=2 的情境下，AP0 的特征 `ap_from_ap_1_sum_ant_rssi` 与 AP1 的特征 `ap_from_ap_0_sum_ant_rssi` 合并为 `ap_from_ap_other_sum_ant_rssi`。合并示意如图 4.9，具体合并规则请见附录 A。

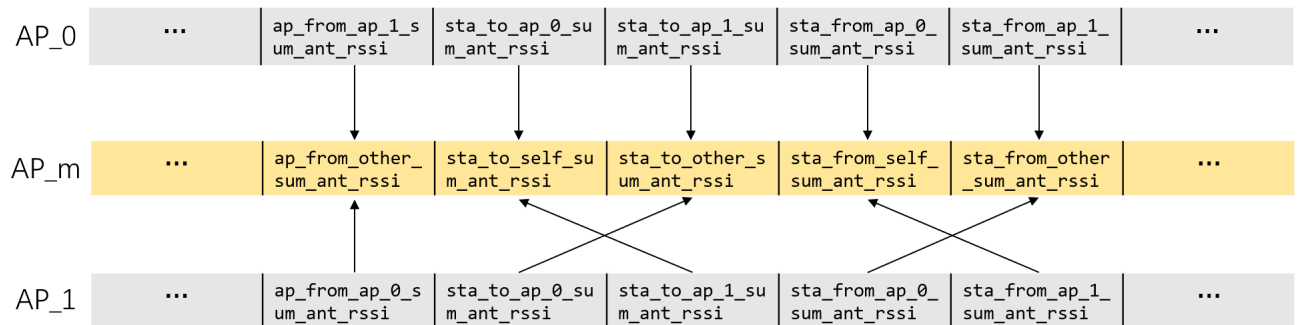


图 4.9 AP 子表合并示意图

4.2.2 各参数影响程度排序

考虑到目标预测数据 `seq_time` 是连续变量，并且为了充分求解特征变量对目标变量的重要性，我们采用了建模方法：随机森林。

随机森林 (Random Forest):

随机森林是一种集成学习方法，基于决策树构建多个分类或回归模型，然后通过集成多个模型的结果来提高模型的准确性和稳定性。它特别适用于高维数据和非线性数据，具有较强的抗过拟合能力。随机森林的基本思想是通过构建多个决策树，并结合它们的预测结果来做最终决策。其主要特点是通过随机性来减少模型的方差，从而避免过拟合。

随机森林的工作原理是：从原始数据集中随机有放回地抽取多个子集（即 **Bootstrap** 样本），每个子集大小与原始数据集相同，但每个样本可能在某个子集中出现多次。对每个子集分别训练一棵决策树。在构建每棵决策树时，随机选取一部分特征进行分裂，避免所有树使用相同的特征进行建模。这一过程引入了随机性，称为特征随机性。• 由于每棵树都是在不同的数据子集和特征子集上构建的，所以每棵树都是不同的。对于回归问题，最终的预测值是所有树预测值的平均。

随机森林具有很强的抗拟合性，特别是在数据噪声较多的情况下。观察到本题训练集中变量较多，可推断的具有较强影响的变量较少，结合随机森林的高维数据的处理能力和可解释性，初步使用该模型是个稳健的选择。

通过随机森林模型输出的各参数影响程度排序如下：

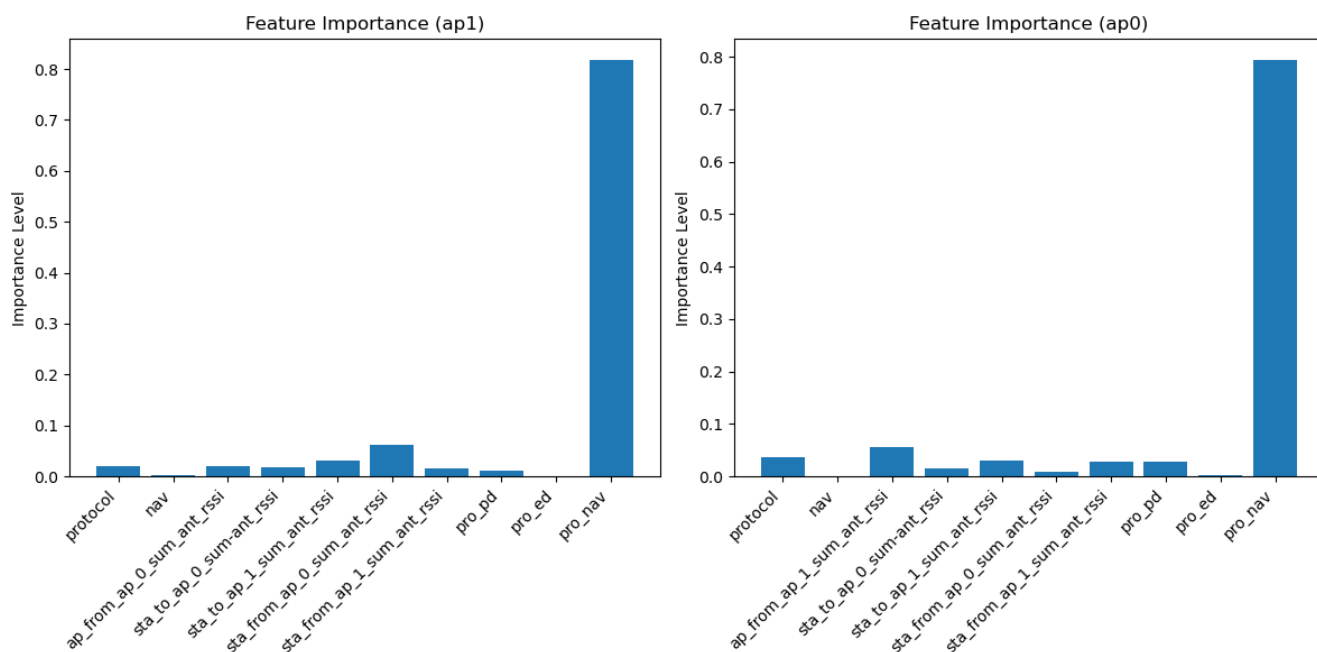


图 4.10 2AP 情况下各参数影响程度排序

我们自己人工根据 `ap_from_ap_other_mean_ant_rssi` 等指标与 `nav` 门限所创造的

指标	相关系数
pro_ed	0.0204786
nav	0.00224272
pro_pd	0.01946922
sta_from_ap_1_sum_ant_rssi	0.01885899
sta_to_ap_0_sum_ant_rssi	0.03116345
ap_from_ap_0_sum_ant_rssi	0.06179776
protocol	0.01493922
sta_to_ap_1_sum_ant_rssi	0.01228365
sta_from_ap_0_sum_ant_rssi	0.00084504
pro_nav	0.81792136

表 4.3 各特征变量与相关系数 - AP1

指标	相关系数
nav	0.03613917
pro_ed	0.0009684
sta_from_ap_0_sum_ant_rssi	0.05637722
sta_to_ap_0_sum_ant_rssi	0.01610746
pro_pd	0.02931114
sta_from_ap_1_sum_ant_rssi	0.0089415
sta_to_ap_1_sum_ant_rssi	0.02875819
protocol	0.02802079
ap_from_ap_1_sum_ant_rssi	0.00144091
pro_nav	0.79393525

表 4.4 各特征变量与相关系数 - AP0

pro_nav 最重要，体现了传输方式；其次是 *sta_from_ap_other/self_sum_ant_rssi* 与 *protocol* 最重要，最后是 *nav* 门限与根据 *ed* 门限和 *ap_from_ap_other_max_ant_rssi* 等特征，以及 *sta_from_sta_0/1* 等指标。

4.2.3 模型建立

建立预测模型，我们分别建立了随机森林、XGBoost、Ridge 模型进行 *seq_time* 的预测。采用交叉验证法评古模型性能。

XGBoost: 对于包含 n 条 m 维的数据集，XGBoost 模型的预测结果可表示为：

$$\hat{y}_i = \sum_{k=1}^K f_k(x_i), \quad f_k \in \mathcal{F}, \quad i = 1, 2, \dots, n$$

其中， $\mathcal{F} = \{f(x) = w_{q(x)}\}$ ， $q: \mathbb{R}^m \rightarrow \{1, 2, \dots, T\}$ 是决策树的结构，表示输入样本被映射到对应叶子节点的路径， T 为叶子节点的数量， $w \in \mathbb{R}^T$ 代表叶节点上的得分值。决策树的结构集合 \mathcal{F} 中的每个函数 f_k 都表示一棵 CART 决策树。

一般来说，损失函数用于衡量模型预测值 \hat{y}_i 与真实值 y_i 之间的差异。对于 n 个样本，采用平方损失函数可表示为：

$$L = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t)})$$

进一步，目标函数（Objective Function）可以写为：

$$\text{Obj}^{(t)} = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t)}) + \Omega(f_t)$$

其中， $\Omega(f_t)$ 表示基模型的复杂度度量，若基模型为决策树，树的深度和叶子节点数等因素都会反映其复杂度。这一正则化项有助于防止模型过拟合。

岭回归（Ridge Regression）：在特征变量的相关性分析中，发现许多变量之间存在显著的线性关系。虽然已进行了一定的特征筛选，但仍可能存在多重共线性问题。为应对这一挑战，拟采用岭回归（Ridge Regression）进行检查与预测。岭回归是一种对线性回归模型的扩展，旨在通过正则化降低因多重共线性导致的过拟合现象。其核心在于向损失函数引入 L2 正则化项，以控制模型复杂度，从而提高泛化能力。

当特征高度相关时，线性回归的系数可能会变得不稳定，导致模型在训练集上表现优异但在测试集上效果不佳。岭回归通过优化损失函数：

$$\text{损失函数} = \sum (y_i - X_i\beta)^2 + \lambda \sum \beta_j^2$$

其中， λ 为正则化参数，用以调控惩罚项的影响。正则化项 $\sum \beta_j^2$ 旨在限制回归系数的大小，防止模型复杂度过高。

当 $\lambda = 0$ 时，岭回归等价于普通线性回归；随着 λ 的增大，回归系数会逐渐缩小，模型复杂度相应降低。

通过这种方式，岭回归有效缓解了多重共线性带来的问题，并提升了模型在新数据上的表现。

4.2.4 结果对比

对于模型预测结果，采用 K 折交叉验证 (k-fold cross-validation) 的方法将数据集划分成 K 等分，依次取每一份作为测试集，剩下的 K - 1 份为训练集。交叉验证重复 K 次，取 K 次准确率的平均值作为最终模型的评价指标。在我们的验证中，K 值取 10，分别对三个模型的预测结果进行了评估。结果如表4.5所示。

由表上表数据对比可知，随机森林模型的准确率最高，因此我们选择随机森林模型对 test 表进行预测。根据题目要求对测试集 test_set_1_2ap 和 test_set_1_3ap 中 AP 发送数据帧序列的总时长进行预测，并填写在对应数据位中。

模型	最高 R ² 系数	交叉验证
随机森林	0.96	0.87
XGBoost	0.94	0.87
岭回归	0.94	0.87

表 4.5 三种模型准确率对比

5 问题二的分析与求解

5.1 问题分析

5.1.1 问题简述

问题二主要研究 AP 在发送过程中选用的调制编码方案 (Modulation and Coding Scheme, MCS) 和空间流数 (Number of Spatial Stream, NSS) 表征。题目中要求, 通过分析实测训练集中的测试基本信息, 特别是节点间 RSSI 信息和门限信息进行分析, 针对测试中 AP 发送数据选用最多次数的 (MCS, NSS) [2] 进行建模, 使用得到的预测模型预测 test_set_2_2ap 和 test_set_2_3ap 中的 (MCS, NSS)。

“选用最多次数的 (MCS, NSS)” 理解: 经观察, 训练集中每一行测试数据都给定了一组 (MCS, NSS), 我们认为这是 AMC 算法在测试初始阶段快速收敛后所得到的, 并在余下的大部分测试期间都使用这一组 (MCS, NSS) 用于解调信号, 即选用最多次数的 (MCS, NSS)。

通过得到这一数值对 (MCS, NSS), 可以一定程度上表示 AP 发送数据的 PHY 层速率 (PHY Rate), 其可以作为求解第三问系统吞吐量的前置影响条件。

5.1.2 机制分析

在 WLAN 网络中, AP 的调制编码方案 (MCS) 和空间流数 (NSS) 是影响物理层速率的重要因素。MCS 表示调制方式和编码速率的组合, NSS 表示数据可以通过多少空间流同时传输。这两个参数会影响 AP 的 PHY 层速率 (PHY Rate), 从而影响数据传输的效率。

通常情况下, AP 根据信道条件 (如 RSSI 和信干噪比 (SINR)) 以及网络环境动态调整 MCS 和 NSS。其调节过程通过自适应调制编码 (AMC) 算法完成。在较好的信道条件下, AMC 算法会选择高阶调制方案和较高的 NSS, 以提高传输速率; 而在信道条件较差时, AMC 算法会选择低阶调制和较低的 NSS, 以保证传输的可靠性。

因此, 预测 AP 的 MCS 和 NSS 实际上是通过了解信道质量以及 AP 的传输环境, 模拟 AMC 算法的决策过程。这一过程受 RSSI、SINR、CCA 门限、NAV 机制等影响。

5.1.3 求解思路

针对上述问题，提出如下求解思路：

由题目可知，AP 发送数据时的 PHY 层速率 PHY Rate 取决于调制编码方案 MCS 和空间流数 NSS，而这两者（MCS, NSS）又由信号解调时的 SINR 决定。WLAN 采用 AMC 自适应调制解调算法，平衡当前 SINR 下所支持的（MCS, NSS）数值大小，以保证丢包率不高于一定阈值。因此，处理本题时，需要通过测试信息，初步计算信道 SINR 信息，以进一步预测对应的（MCS, NSS）。

AP 的 AMC 所选用的（MCS, NSS）不仅与 SINR 相关，同时也与 AP 的传输方式相关。可以总结的，两个 AP 之间的传输方式有三种可能：同步传输、异步传输、同步异步混合传输。在各个传输方式下，因为 STA 接收数据时受到的干扰来源不同，导致 SINR 的计算方法也不同。因此，为了计算信道 SINR 信息，需要确定各测试数据所处的传输方式，而这一过程需结合问题一中使用节点间 RSSI 信息和门限信息进行分类的方法。

为解决问题二，我们首先根据节点间 RSSI 信息和门限信息对其所处**传输方式**进行分类，随后根据其实际传输方式用不同的计算方式得到对应 **SINR 信息**，进一步地，利用 SINR 对（MCS, NSS）数值进行预测。本题的数据处理过程仍然沿用第一题的方法，但是去除了主表拆分和合并的过程，这是因为我们发现分表后，子表之间的特征分析具有很大相似性，故直接使用全部数据进行分析。

在预测过程中，我们发现，尽管预测得到的 MCS 值具有很大的波动性，但是 NSS 值相对稳定，维持在 NSS2 附近。这一表现难以评价模型的准确性和拟合效果，因此，我们改为预测 MCS 和 PHY Rate 值，评价模型预测效果，随后利用文中提到的（MCS, NSS）与 PHY Rate 对应关系，得到对应的 NSS，填写预测结果。

问题二思路如图5.11：

5.2 问题求解

5.2.1 数据处理

在第一问处理的基础上，本问题进一步确定了因为 AP 间 RSSI 所处门限区间，用以分类其在一次测试中的主要传输方式，以此进一步计算 SINR 数值。

5.2.2 模型构建

由题目得知，无线信道具有瞬息万变的特点，实测中所测量的 RSSI 信息属于大尺度信息，不足以完全反应真实信道变化，因此，我们使用多层感知机（MLP, Multilayer Perceptron）辅助进行目标值预测。

多层感知机（MLP, Multilayer Perceptron）：MLP 是一种经典的前馈神经网络，由输入层、一个或多个隐藏层以及输出层构成。MLP 是一种全连接网络，即每一层中的每个神

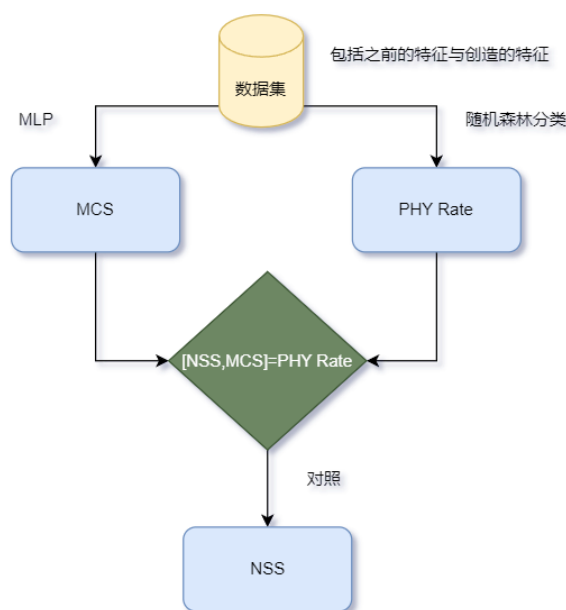


图 5.11 问题二思路示意图

经元都与下一层的所有神经元相连，主要用于处理结构化数据以及分类和回归任务。

- 输入层操作：我们对数据进行标准化操作，输入数据的维度决定了输入层神经元的数量，此外输入层不进行任何其他计算，功能是将输入数据传递给下一层。
- 隐藏层操作：隐藏层在 MLP 中担任核心角色，其中包含了多个神经元，并通过线性变换与非线性激活函数对输入数据进行处理。隐藏层与上一层所有神经元紧密相连，并从输入数据中提取复杂的模式和特征。该模型中前向传播用于将输入数据通过每一层神经网络，最终产生输出。反向传播则通过计算输出与真实标签的误差（损失函数）来更新权重和偏置，常用的优化算法是梯度下降和 Adam。在此我们使用了两层隐藏层，以充分探索数据集中的复杂关系，并使用 Adam 优化器来对模型进行训练。
- 激活函数：

- ReLU (Rectified Linear Unit)：常用于隐藏层，ReLU 函数定义为

$$f(x) = \max(0, x)$$

它能够引入非线性，同时避免梯度消失问题。

- Sigmoid：适用于概率预测，输出值在 (0, 1) 之间。
- Tanh：将输入压缩到 (-1, 1) 范围，常用于对数据中心化的任务。

在该模型中，我们使用了 ReLU 激活函数，因为最后预测结果并不需要对输出范围进行限制。

- 前向传播与反向传播：MLP 的训练过程基于前向传播和反向传播。

- 正则化技术：为防止过拟合，MLP 中常采用正则化技术，例如 Dropout 和 L2 正则化。Dropout 随机丢弃一部分神经元的连接，减少模型对特定权重的依赖，提高模型的泛化能力。在模型中我们针对隐藏层的输出结果进行了 0.2 概率的 Dropout。

5.2.3 结果分析

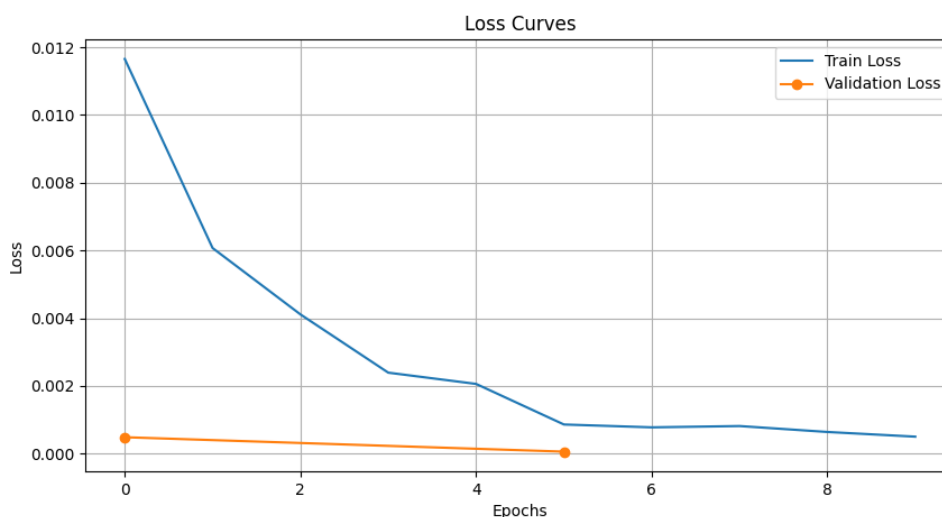


图 5.12 MLP 模型训练过程

由上图可知，模型在训练集和验证集上的损失值逐渐下降，且在验证集上的损失值与训练集上的损失值保持一致，说明模型具有较好的泛化能力。但是因为用于训练的数据集很小，所以仅训练了较少的 epoch。

同频 AP 个数为 2:

使用 10 折交叉验证对预测结果的 PHY Rate 进行评估，得到的 R2 系数均值为 0.89，10 次测验的值分别为:[0.91014529, 0.97066302, 0.83193871, 0.95018807, 0.96852579, 0.75103629, 0.94168775, 0.80378873, 0.95456839, 0.83780847]。

使用 10 折交叉验证对预测结果的 MCS 进行评估，得到的 R2 系数均值为 0.91，10 次测验的值分别为: [0.84210526, 0.92105263, 0.89473684, 0.92105263, 0.89473684, 0.92105263, 0.86842105, 0.94736842, 0.92105263, 0.97297297]。

同频 AP 个数为 3:

使用 10 折交叉验证对预测结果的 PHY Rate 进行评估，得到的 R2 系数均值为 0.85，10 次测验的值分别为:[0.91701557, 0.86328318, 0.87501772, 0.86086456, 0.84691466, 0.80411334, 0.82224948, 0.84631725, 0.84689186, 0.88545755]

使用 10 折交叉验证对预测结果的 PHY Rate 进行评估，得到的 R2 系数均值为 0.86，10 次测验的值分别为:[0.91701557, 0.86328318, 0.87501772, 0.86086456, 0.84691466, 0.80411334, 0.82224948, 0.84631725, 0.84689186, 0.88545755]

上述结果证明模型预测较为准确，且通过观察 10 次结果可知，模型具有较好的鲁棒性。

如 6.1.3 节末尾处提到，此处没有评估和预测 NSS 值，因为预测结果的 MCS 值具有很大的波动性，但是 NSS 值相对稳定，维持在 NSS2 附近。这一表现难以评价模型的准确性和拟合效果，因此，我们改为预测 NSS 和 PHY Rate 值，评价模型预测效果，随后利用文中提到的 (MCS, NSS) 与 PHY Rate 对应关系，得到对应的 MCS，填写预测结果。

6 问题三的分析与求解

6.1 问题分析

6.1.1 问题简述

问题三主要研究 WLAN 网络系统吞吐量预测。题目要求结合问题一和问题二的分析，利用前两问的预测指标，包括 AP 发送数据帧序列的总时长、(MCS, NSS) 以及其他测试基本信息，共同构建预测模型，通过测试集 test_set_1_2ap 和 test_set_1_3ap 预测网络吞吐量。其中问题二所预测得到的 (MCS, NSS) 无法获得很高精度，允许采用实测中统计的数据帧真实 (MCS, NSS) 作为模型输入变量。

问题三是本课题的最终研究目标，前两问通过分析 WLAN 网络拓扑、节点间 RSSI、信道接入机制、干扰等因素对 WLAN 数据发送、速率的影响，本问题进一步地利用其预测结果作为重要影响因素与支撑，对 WLAN 系统吞吐量进行精确预测。基于该吞吐量预测模型，对 WLAN 进行优化，有望突破工业、教育、医疗等新场景，为用户提供极致的业务体验。

6.1.2 机制分析

根据题目背景，给出如下定义：

- 吞吐量 (throughput)：吞吐量指节点单位时间内成功发送的比特数。仅 MSDU 的总字节数是有效传输数据，进行吞吐量计算。
- 丢包率 (per)：发送数据帧的失败个数与总个数的百分比。
- 数据的帧序列的总时长 (seq_time) (us)：所有成功或失败的数据帧发送，帧序列时长从开始发送 RTS 到收到 ACK，或者超时。
- 数据帧的时长 (T_{PPDU}) (s)：一个数据帧的时长。一次测试里，统计每个数据帧的时长，取平均值。仅包括 PPDU 的传输时间，不包括 RTS, ACK 等。
- AMPDU 个数 (N_{AMPDU})：一个数据帧的聚合个数。一次测试里，统计每个数据帧的聚合个数，取平均值。聚合的详细解释见第 4 节。
- TCP 的 ACK 算是数据帧，算作吞吐量，个数是下行数据帧的 3:1 到 2:1 左右。根据丢包率改变传输速率。ACK 的大小为 64Bytes。UDP 仅计算 AP 发往 STA 的数据帧，发送时间间隔服从泊松分布，发送数据的速率为 290 Mbps。

(1) 数据包长度分析：

设 N_{success} 为一次时长 60s 的测试中成功发出的数据包个数。 N_{timeout} 为测试中失败的包的个数，有：

$$T_{\text{seq_time}} = N_{\text{timeout}} \times T_{\text{timeout}} + N_{\text{success}} \times T_{\text{success}} \quad (6.4)$$

其中, T_{success} 为成功发送数据帧的时长, 即 T_{PPDU} ; T_{timeout} 为失败发送数据帧的时长, 根据6.13可知, 传输失败时时长包括 RTS 时长与超时时长, 即 $T_{\text{timeout}} = T_{\text{RTS}} + T_{\text{ACK_timeout}} = 85 \mu\text{s}$ 。

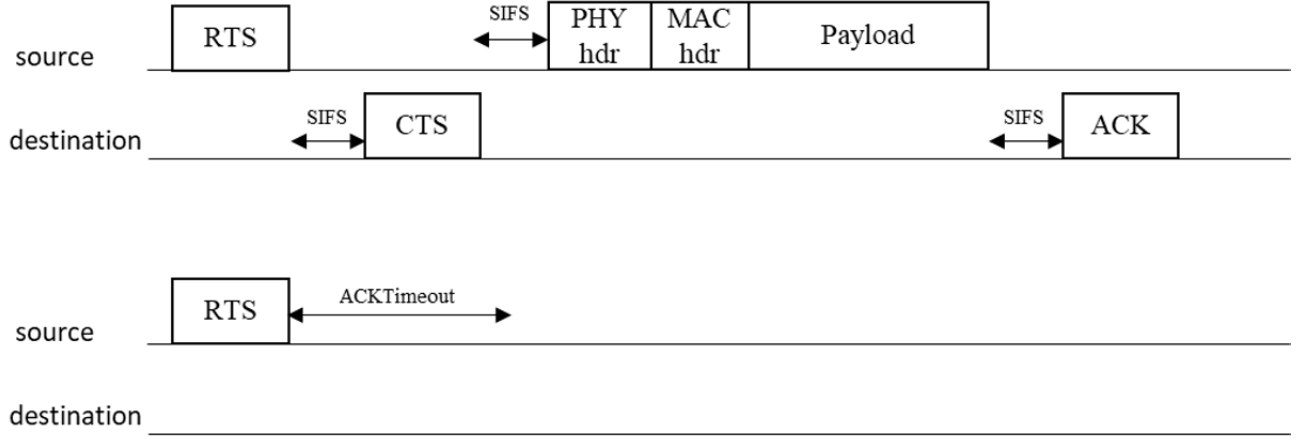


图 6.13 RTS-CTS 模式下发送成功发送失败的帧序列

又由丢包率定义可知, $\text{PER} = \frac{N_{\text{timeout}}}{N_{\text{success}} + N_{\text{timeout}}}$, 变换得:

$$N_{\text{timeout}} = \frac{\text{PER}}{1 - \text{PER}} \times N_{\text{success}} \quad (6.5)$$

因此, 由式6.4与式6.5可得:

$$N_{\text{success}} = \frac{T_{\text{seq_time}}}{T_{\text{success}} + \frac{\text{PER}}{1 - \text{PER}} \times T_{\text{timeout}}} \quad (6.6)$$

记测试中一个 PPDU 数据包的长度为 L_{PPDU} 对于吞吐量 Throughput(MBps), 有,

$$\text{Throughput} = \frac{N_{\text{success}} \times L_{\text{PPDU}} \times 8.0}{60 \times 10^6} \quad (6.7)$$

即,

$$L_{\text{PPDU}} = \frac{\text{Throughput} \times 60 \times 10^6}{N_{\text{success}} \times 8} \quad (6.8)$$

综上所述, 在训练集中, 吞吐量已给出的情况下, 我们可以利用吞吐量、丢包率和数据帧发送总时长, 计算出测试中聚合数据包 PPDU 的平均长度。

(2) 聚合参数分析:

为了提升发送小包的效率, 协议允许通过聚合一次发送多个具有相同目的地址的数据包 [3], 如6.14。在一次数据包聚合 (AMSDU) 过程中, 多个相同接收地址的同服务类别的 MAC 服务数据单元 (MSDU) 封装成一个 MAC 协议数据单元 (MPDU)。组装好的多

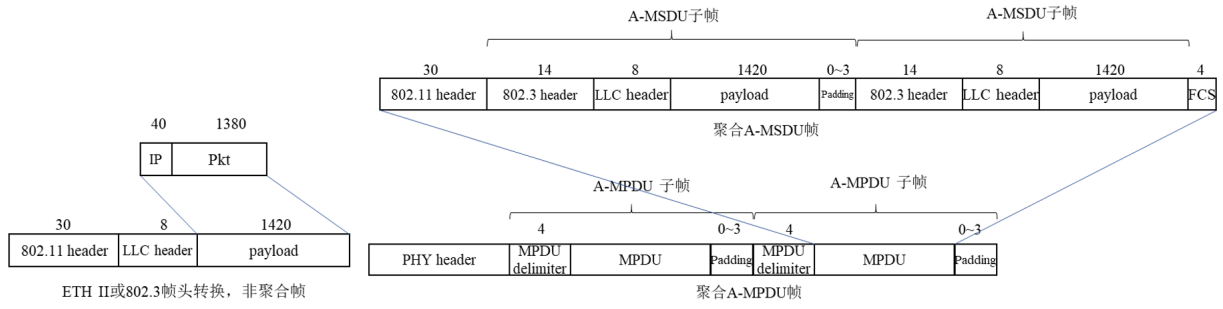


图 6.14 聚合数据包示意图

个 MPDU 进一步聚合（AMPDU），这一过程包括将多个 MPDU 聚合成一个 PHY 协议数据单元（PPDU）。此处 PPDU 即为在 WLAN 网络中进行传输的一个数据帧。

组成 MPDU 的 MSDU 包长计算如式6.9（单位：Bytes）：

$$L_{\text{MSDU}} = L_{\text{802.3header}} + L_{\text{LLCheader}} + L_{\text{payload}} + L_{\text{padding}} \quad (6.9)$$

其中，802.3header 为 14Bytes，LLCheader 为 8Bytes，此处 payload 即为测试基本信息中的‘pkg_len’，长度为 1500Bytes。AMSDU 过程中需插入 0-3 字节的填充字节进行 4 字节对齐，此处 padding 计算可得为 2Bytes。

组成 PPDU 的 MPDU 包长计算如式6.10（单位：Bytes）：

$$L_{\text{MPDU}} = L_{\text{802.11header}} + L_{\text{MSDU}} \times N_{\text{AMSDU}} - L_{\text{padding}} + L_{\text{FCS}} \quad (6.10)$$

其中，802.11header 为 30Bytes，FCS 为 4Bytes。

最终，计算一个完整数据帧 PPDU 的包长（单位：Bytes）式6.11：

$$L_{\text{PPDU}} = (L_{\text{delimiter}} + L_{\text{MPDU}} + L_{\text{padding}}) \times N_{\text{AMPDU}} + L_{\text{PHY}} \quad (6.11)$$

其中， $L_{\text{delimiter}}$ 为 4Bytes，PHY 为 PHY 头部大小。在 802.11 WLAN 中 [4, 5]，最常见的 PHY 头部大小通常是 160 位（20 字节），适用于 802.11a、802.11g、802.11n、802.11ac 和 802.11ax 等标准。这是因为这些标准大多数都基于 OFDM（正交频分复用）调制，采用相同的 PHY 头部结构对于 802.11b（DSSS 和 CCK），虽然它的 PHY 头部是 192 位（24 字节），但在现代无线网络中，802.11a/g/n/ac/ax 的使用更为广泛，因此 160 位的 PHY 头部是最常见的。此外，不同的 PHY 头字节数差异不大，我们认为对最终结果的造成误差可以忽略不计。

已知 PPDU 长度（即 L_{PPDU} ），且

$$L_{\text{padding}} = 4 - L_{\text{MPDU}} \% 4 \quad (6.12)$$

则 L_{MPDU} 可以根据如下计算得出，联立6.12与6.11，带入已知数值：

$$L_{\text{MPDU}} - L_{\text{MPDU}} \% 4 = \frac{L_{\text{PPDU}} - 20}{N_{\text{AMPDU}}} - 8 \quad (6.13)$$

由 $L_{\text{MPDU}} \% 4 \in [0, 3]$ ，可以认为

$$L_{\text{MPDU}} = \frac{L_{\text{PPDU}} - 20}{N_{\text{AMPDU}}} - 8 + r, \quad r \in [0, 3] \quad (6.14)$$

由于 MPDU 实际值很大（几千 bytes），则不妨取 $r=0$ ，又因为一个 MSDU 的长度为 1524bytes（式6.9），可得：

$$L_{\text{MPDU}} = 32 + 1524 \times N_{\text{AMSDU}} \quad (6.15)$$

联立6.14和6.15两式可得：

$$N_{\text{AMSDU}} = \frac{\left(\frac{L_{\text{PPDU}} - 20}{N_{\text{AMPDU}}} - 40 \right)}{1524} \quad (6.16)$$

综上，在已知 L_{PPDU} 的情况下，我们可以得到 N_{AMSDU} 和 N_{AMPDU} 之间的关系表达式，由于训练集中没有给出 AMSDU 聚合包个数 N_{AMSDU} 的对应值，我们可以通过训练集中给定的吞吐量等数据，通过式6.8计算出聚合数据包 PPDU 的平均长度 L_{PPDU} ，随后利用同样给出的 AMPDU 聚合包个数 N_{AMPDU} ，通过式6.16计算出 AMSDU 聚合包个数。

6.1.3 求解思路

根据题目可知，WLAN 网络吞吐量主要取决于 AP 发送机会、发送时所选用的 PHY Rate 以及数据帧的比特数。其中，AP 发送机会即为问题一中的 `seq_time`，PHY Rate 可以根据下表映射到问题二中所预测的（MCS, NSS）。本问题中，主要通过分析数据帧比特数对吞吐量的影响。

吞吐量指节点单位时间内成功发送的比特数，而有效传输数据仅包括 MSDU 的总字节数。因此，我们认为，深度挖掘聚包机制是具有意义的，其中使用的参数在预测系统吞吐量过程中，具有较高参考价值，其中，数据帧的时长 T_{PPDU} 、AMSDU 聚合个数 N_{AMSDU} 和 AMPDU 聚合个数 N_{AMPDU} 参数被认为具有代表性。然而通过观察测试集发现，上述几个变量均未给出。因此我们希望先通过类似一、二两问的预测过程，先得出 T_{PPDU} 、 N_{AMSDU} 和 N_{AMPDU} 的预测值，并用于新的模型输入，用来预测吞吐量。

问题三思路如图6.15：

6.2 问题求解

6.2.1 模型构建

本问题中，先用多层感知机 MLP 模型预测出 T_{PPDU} 和 N_{AMPDU} 作为预测值，随后，使用 XGBoost 模型预测得到吞吐量。此处，根据散点图6.16可知， N_{AMSDU} 与 N_{AMPDU} 呈现正相关，系数为 0.6，且相关性较高，故只考虑 T_{PPDU} 和 N_{AMPDU} 作为特征变量。

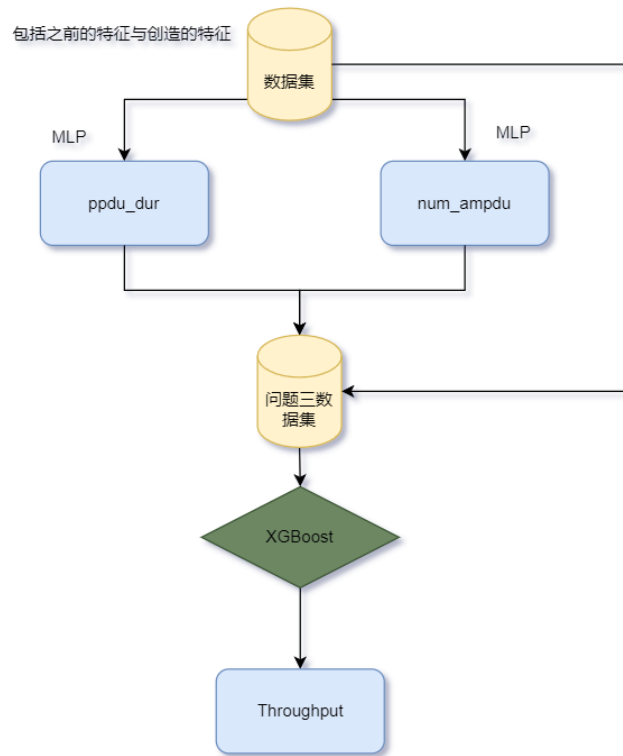


图 6.15 问题三思路示意图

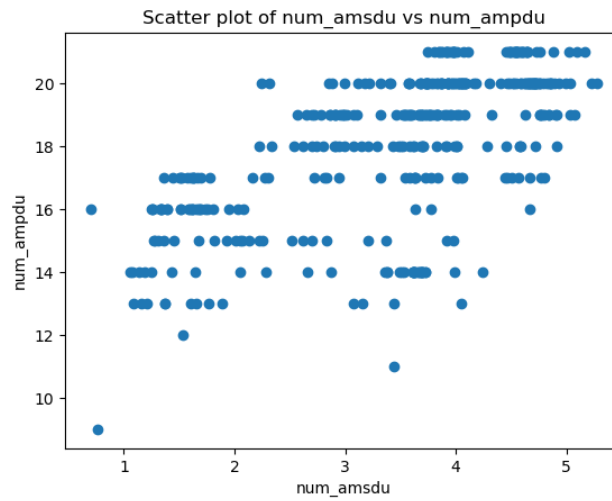


图 6.16 N_{AMSDU} 与 N_{AMPDU} 关系展示图

6.2.2 结果分析

问题 3 中 ap2 对于 throughput 的值回归,均值为 0.96,10 折验证结果为 `array([0.97346603, 0.96983368, 0.97503999, 0.93833553, 0.93052316, 0.98189316, 0.97710122, 0.96557072, 0.94913004, 0.95445993])`

问题 3 中对于 ap3 对于 throughput 的值的回归，均值为 0.95，10 折交叉验证结果为 array([0.97497345, 0.98397, 0.95935381, 0.97253042, 0.97065528, 0.95160585, 0.97799428, 0.95418978, 0.96758076, 0.9685466])

由结果可知，模型预测效果良好，且鲁棒性较强。

7 模型总结

7.1 方案优点

本文在充分 WLAN 机制的基础上，对数据集进行了特征创造，用概率的方式体现同频 AP 之间通信的模式。该人工特征在后续的特征重要程度排名中名列前茅，证明了该人工特征的合理性与有效性。

本文从一开始解决问题的时候，就利用最大条件引入了后续可能用到的特征，如 SINR，聚类状态等信息，并采用多个模型和算法进行分析，最大程度上利用数据集中的信息。

本文对于所有题目中 2AP 与 3AP 的情况进行了分开讨论，针对容易模拟的 2AP 情形，对使用的模型进行横向对比，至于情况复杂多变的 3AP 情形，使用多层感知机来探究其复杂关系。

在针对包聚合问题的时候，采用机理分析与深度学习预测相结合的方式，填补了其中所需要的缺失参数，并成功对目标值进行预测。

7.2 方案不足

用来给深度学习模型训练的数据集的数量并不是很多，深度学习模型预测的参数效果可能会差强人意，同时基于预测值所分析的结果质量也会下降；

模型如 MLP，XGBoost 等模型预测性能虽然很好，但是解释性相对较差，若要在相关领域进行预测分析，可能需要改进并更新对应的模型解释工具。

7.3 未来的改进方向

获得更多的真实数据，并将基站的位置信息等多方面特征因素纳入考虑，提高模型的鲁棒性和稳定性；

可以考虑模型融合技术或启发式算法优化模型技术，将多个模型的预测结果进行融合，或者对模型效果进行优化。

参考文献

- [1] 方震, 赵湛, 郭鹏, 等, 基于 RSSI 测距分析, 传感技术学报, 20(11):2526-2530, 2007.
- [2] T.R. P, T.H. K, J.Y. C, et al., Throughput and energy consumption analysis of IEEE 802.15.4 slotted CSMA/CA, Electronics Letters, 41(18):1017-1019, 2005.
- [3] 无线局域网中发送 PPDU 的方法及设备.
- [4] 罗振东, 下一代 WLAN 技术标准 802.11ac/ad, 现代电信科技(12):10-14, 2010.
- [5] Alabady S A, Salleh M F M, Hasib A, Throughput and Delay Analysis of IEEE 802.11 DCF in the Presence of Hidden Nodes for Multi-hop Wireless Networks, Wireless personal communications: An International Journal, 79(2):907-927, 2014.

8 附录 A 题目测试集结果

表 8.6 test_set_1_2_ap

Predict Seq Time	Predict Throughput
47.2651	124.665
48.3345	133.4773
49.8563	141.9504
50.1373	156.6426
47.2461	129.1914
48.5465	139.6945
49.1479	141.9945
50.2585	159.6448
47.2017	129.3225
48.0187	137.0568
49.0428	144.9029
50.4356	156.3343
47.7852	128.562
48.1161	147.953
49.6241	140.187
50.2939	146.1679
47.3202	150.0952
48.1482	128.1342
49.2009	163.1809
50.336	165.8982
47.8155	145.9913
48.1552	146.455
49.4027	165.9966
50.3686	141.2401
47.7371	146.6508
48.3117	169.0632
49.0145	180.0328
50.2669	184.9271
47.1948	150.6723
47.9691	154.1246

Predict Seq Time	Predict Throughput
49.3971	177.8843
50.0384	179.9314
46.2543	164.0636
48.2259	158.411
48.7257	182.3656
50.2888	192.1225
41.9092	150.6678
48.2105	173.6938
44.7022	158.3659
49.9435	186.3889
34.9224	131.2143
46.7297	154.9227
36.7786	137.9985
49.1878	175.1451
35.1717	129.7112
44.3223	160.4146
38.1973	147.8302
48.3467	132.1562
34.1428	143.9704
40.2908	84.4042
36.131	150.7141
45.1864	106.31
28.8219	136.6747
40.0162	83.4793
30.1281	147.7851
45.2789	96.9388
29.5895	134.8225
31.762	95.4797
32.806	153.1011
33.0985	98.1278
28.9542	139.0358
24.2376	83.0778
30.9204	148.2933

Predict Seq Time	Predict Throughput
27.1482	88.6441
33.8094	145.8498
23.6982	85.8838
29.3689	141.2124
23.5351	84.7705
31.3739	152.7492
26.2432	88.7686
28.3373	138.2543
24.1125	88.368
29.8817	151.4965
25.409	92.314
27.6467	129.4512
23.6852	75.2333
26.9599	134.9168
23.414	86.954
29.6329	145.4149
24.6413	91.1731

表 8.7 test_set_1_3_ap

Predict Seq Time	Predict Throughput
25.0064	23.3511
37.4308	38.4647
41.4869	52.7878
32.53	33.9107
36.4835	37.5621
37.2803	35.3279
24.0461	25.4492
42.0029	53.5889
41.0497	52.4553
21.903	26.9922
42.2178	52.8659
41.6774	50.0746

Predict Seq Time	Predict Throughput
30.7003	38.0648
39.532	49.4355
35.5419	34.2947
23.7177	26.3978
39.2109	47.8945
41.405	54.3591
22.7145	48.5024
44.1882	59.132
41.9338	50.4032
25.5747	55.6519
44.0885	67.5784
37.0978	34.2534
20.2587	42.5505
39.7868	58.9801
38.9722	91.9616
24.652	54.5131
36.9085	70.756
33.0503	82.0039
23.6014	68.1198
41.4522	58.5663
25.1013	87.1511
24.837	70.3053
40.4583	65.1086
29.3328	54.14
23.3822	67.4068
36.5983	56.4485
26.6571	86.1911
25.8175	57.8042
33.6015	49.6523
30.3998	84.6175
25.0081	58.6933
36.601	39.9178
24.1975	67.3314

Predict Seq Time	Predict Throughput
26.5525	73.7755
33.0559	49.8761
22.2881	58.4201
26.8027	57.1233
32.6059	55.6853
30.2375	81.7713
27.205	77.3873
29.9736	47.8943
23.3535	69.7504
27.6127	61.2838
30.8926	47.7008
24.8584	68.4584
22.7993	48.1896
32.0543	87.9842
22.437	60.3641
22.41	48.6827
25.8643	50.0561
23.8022	75.7764
22.1196	60.714
25.4878	67.5494
21.0649	68.869
22.1639	61.8667
23.5779	67.1891
22.5061	66.8044
23.6727	55.5947
26.5288	60.5337
23.9582	64.0304
21.7973	61.7426
23.7318	66.992
22.7815	66.1026
21.9857	63.9706
24.208	67.7197
22.8256	68.0849

Predict Seq Time	Predict Throughput
22.5585	43.1276
26.0471	62.9898
24.9793	64.1981
20.1463	60.1995
23.4939	68.5515
23.0938	67.8585
22.3694	42.795
26.1427	39.0985
23.3986	66.8948
20.6397	62.1902
23.3422	67.9997
22.7726	67.2747
22.6451	44.8699
24.8425	58.0489
24.4456	70.1642
21.524	63.8814
23.8503	70.0541
22.6385	67.6238
22.5524	49.043
26.1925	60.826
23.8125	62.6679
21.2791	61.8713
23.116	67.2642
22.1655	66.8284
21.372	40.4767
25.1866	39.7041
23.6661	69.6283

表 8.8 test_set_2_2_ap

seq_time	predict nss	predict mcs	PHY_rate
48.399	2	5	117.648
45.4059	2	4	78.353

seq_time	predict nss	predict mcs	PHY_rate
50.0207	2	5	122.292
48.8011	2	4	82.567
48.9702	2	5	118.852
41.0705	2	4	88.3
50.219	2	5	123.84
46.0801	2	4	92.514
47.7481	2	5	116.788
23.1745	2	11	279.111
49.8304	2	5	118.852
22.8541	2	11	277.275
46.8878	2	5	113.348
22.2638	2	11	279.398
49.7409	2	5	115.068
21.762	2	11	277.562
45.3071	2	5	118.194
22.2391	2	11	279.398
44.7828	2	5	122.21
22.1276	2	11	279.398
48.4934	2	5	124.962
22.5802	2	11	279.398
40.4625	1	11	126.828
21.7812	2	11	281.521
39.3801	1	11	123.732
21.6405	2	11	268.38
25.9707	2	11	269.307
21.013	2	11	271.364
27.3625	2	11	264.66
21.3327	2	11	265.396
26.2412	2	11	269.594
21.5822	2	11	271.364
26.3478	2	11	263.799
21.179	2	11	265.396
24.5087	2	11	268.275

seq_time	predict nss	predict mcs	PHY_rate
21.4633	2	11	268.726
27.4484	2	11	252.036
20.5855	2	11	262.758
29.8473	2	11	251.863
21.3745	2	11	268.726
29.8961	2	11	246.068
20.6629	2	11	262.758
30.1124	2	11	251.922
21.3166	2	11	268.726
30.2264	2	11	247.273
21.5307	2	11	268.726
30.4892	2	11	241.478
20.1206	2	11	262.758
30.0842	2	11	251.922
20.018	2	11	256.678
29.8276	2	11	253.241
24.7572	2	11	259.316
29.8292	2	11	253.241
25.7433	2	11	256.678
30.6812	2	11	247.446
28.0749	2	11	241.758
29.697	2	11	247.273
25.7573	2	11	256.678
30.6075	2	11	241.478
28.0807	2	11	241.758
29.774	2	11	247.273
25.77	2	11	256.678
30.5472	2	11	241.478
28.3248	2	11	241.758

表 8.9 test_set_2_3_ap

seq_time	predict nss	predict mcs	PHY_rate
42.2926	2	4	113.872
45.4404	1	9	127.07
44.4092	1	11	138.255
41.7514	2	4	115.936
44.1809	1	8	128.732
43.2099	2	4	139.919
42.1083	2	4	112.838
44.7591	1	10	123.86
38.1024	1	11	135.787
41.2066	2	4	114.558
43.9171	1	10	127.3
33.8255	2	4	137.623
42.6715	2	4	113.183
44.7424	1	9	131.602
38.6409	1	11	140.09
41.5896	2	4	114.215
44.3098	1	9	132.691
34.2283	1	11	141.238
42.9789	2	4	143.211
45.4206	1	11	132.52
39.0837	1	11	141.295
42.0434	2	4	143.557
44.8619	1	8	133.667
34.9088	1	11	141.927
43.5406	2	5	155.786
45.2315	1	11	121.677
37.0907	1	11	141.869
42.5732	2	5	155.788
44.6804	1	8	122.824
32.9818	1	11	142.615
42.4057	2	5	156.642

seq_time	predict nss	predict mcs	PHY_rate
45.462	1	11	122.997
39.1537	1	11	142.098
41.4606	2	5	156.815
44.8034	1	11	123.857
33.9763	1	11	142.5
42.9577	2	5	163.304
44.9166	1	11	124.087
39.4079	1	11	142.098
41.4912	2	6	163.133
44.0732	1	11	125.234
34.8936	1	11	142.156
43.4252	2	5	170.018
44.7666	1	11	122.021
36.0027	1	11	160.806
42.8928	2	7	170.362
43.9838	1	11	123.168
31.943	1	11	162.01
43.6936	2	5	170.363
43.3566	1	11	134.523
34.5452	1	11	160.634
43.1879	2	5	170.363
42.5493	1	11	133.203
31.4775	1	11	162.125
37.1045	2	7	169.101
37.5949	1	11	139.859
30.3536	2	8	186.05
32.0158	2	5	182.454
29.4685	1	11	172.106
29.8212	1	11	202.575
29.9646	2	7	193.761
25.0175	1	11	207.744
29.3631	1	11	202.633
28.1911	2	8	200.071

seq_time	predict nss	predict mcs	PHY_rate
22.0846	2	11	226.964
16.9641	2	11	252.608
28.1997	2	11	225.754
21.6221	1	11	131.086
23.3751	2	8	226.975
28.4066	2	9	217.497
28.9317	1	11	128.159
18.8828	2	8	222.215
15.28	2	11	266.092
18.5784	2	11	272.975
17.7035	2	11	267.234
15.9145	2	11	265.576
18.9978	2	11	265.978
18.1297	2	11	267.808
21.6032	1	11	211.659
23.2012	2	11	234.593
21.9941	2	11	226.504
15.1433	2	11	256.506
18.1183	2	11	276.647
17.5933	2	11	241.817
15.279	2	11	286.8
18.0564	2	11	276.877
17.9117	2	11	241.301

9 附录 B：子表合并规则

表 9.10 2AP 子表变量合并

	AP_0	AP_1	AP_merge
0	protocol	protocol	protocol
1	nav	nav	nav
2	ap_from_ap_1_sum_ant_rssi	ap_from_ap_0_sum_ant_rssi	ap_from_other_sum_ant_rssi
3	sta_to_ap_0_sum_ant_rssi	sta_to_ap_1_sum_ant_rssi	sta_to_self_sum_ant_rssi
4	sta_to_ap_1_sum_ant_rssi	sta_to_ap_0_sum_ant_rssi	sta_to_other_sum_ant_rssi
5	sta_from_ap_0_sum_ant_rssi	sta_from_ap_1_sum_ant_rssi	sta_from_self_sum_ant_rssi
6	sta_from_ap_1_sum_ant_rssi	sta_from_ap_0_sum_ant_rssi	sta_from_other_sum_ant_rssi
7	pro_pd	pro_pd	pro_pd
8	pro_ed	pro_ed	pro_ed
9	pro_nav	pro_nav	pro_nav