SVMDformer：基于物联网中变形器的半监督车辆不当行为检测框架

Zhikang Liu*[1] , Hongyun Xu[1] , Yong Kuang[1] , Feng Li [1]
[1]华南理工大学计算机科学与工程学院，中国广州 202121045561@mail.scut.edu.cn, hongyun@scut.edu.cn,
{csdeaisry, 202221044781}@mail.scut.edu.cn

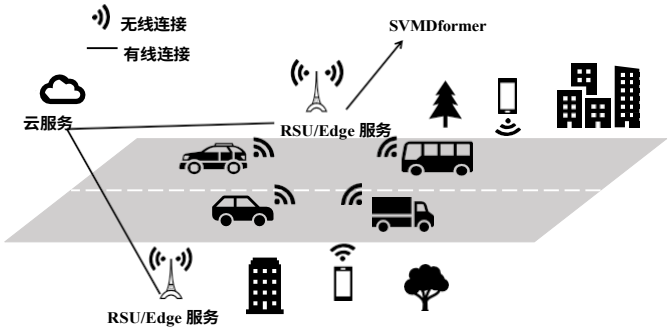
摘要-基于车对物（V2X）的技术已成为解决道路拥堵和交通事故问题的重要研究课题。然而，车联网的快速发展使得大量信息在车联网（IoV）中流动，其中一些信息来自内部攻击者的不当行为，从而威胁到 IoV 的安全。近来，一些作品尝试解决不当行为检测问题，但存在不当行为类型不全、多模型检测效率低、准确性不足等问题。本文提出了一种基于变换器（SVMDformer）的半监督车载不端行为检测框架，它能将车载信息序列变换为不端行为得分，并通过设置阈值来识别不端行为。该框架

SVMDformer 在云服务器中进行训练，并部署在边缘服务器中，用于检测 IoV 中传入车辆信息的不当行为。基于一个开源数据集，我们演示了该模型在不同环境下的性能，分析了其灵敏度，并将其与基线进行了比较。与之前的研究相比，所提出的 SVMDformer 更具通用性、准确性和全面性。它是一个单一模型，可检测车辆信息序列的 19 种不当行为。它的 AUC 达到 99.87%，准确率达到 99.66%，精确度达到 99.68%，F1 分数达到 99.66%，误检率为 0.34%，比最先进（SOTA）的工作低三倍多。

***Index Terms-IoV*, Vehicular Misbehavior Detection, Semi-supervised Learning, Transformer**

I. 导言

根据最近的一份报告[1]，全球对车辆的需求不断增加，交通拥堵和交通事故也随之增加。车联网（IoV）是一种理想但尚未落地的车载网络（如图 1），它与物联网（IoT）、车对万物（V2X）[2] 和移动边缘计算（MEC）[3] 技术相结合。IoV 为车辆提供了便利性和安全性，已成为未来协同智能交通系统（C-ITS）的关键部分[4]。在 IoV 中，每辆车都安装了车载单元（OBU），以便与其他基础设施通信，并成为 IoV 的一部分。联网车辆交换包含



位置、速度、加速度和航向信息的基本安全信息（BSM）。

然而，V2X 系统在带来便利的同时也存在安全问题，包括来自外部和内部的攻击和不当行为。来自外部的攻击可以通过密码学（如公钥基础设施（PKI）[5]）来识别，而内部攻击者则拥有合法的密钥来进行通信。

图 1：IoV 模型。

但它们会发送行为不端的 BSM，干扰道路安全和交通效率。例如，行为不端的内部人员不断发送位置信息，或通过伪造新 ID（Sybil）制造虚假交通拥堵，让用户误以为前方拥堵，导致道路资源浪费。在物联网中发送此类错误信息被称为不当行为，一般分为故障和攻击，检测它们被称为不当行为检测（MD）[6]。

许多工作[7]尝试了基于机器学习和以数据为中心的方法来解决 MD 问题。然而，所有方法都存在问题，如错误行为类型不完整、多模型检测效率低下和准确性不足。传统的机器学习（ML）无法处理 V2X 的海量数据场景。然而，深度学习的发展为 MD 提供了新的途径。半监督学习（Semi-supervised learning）在异常检测（AD）[8]领域被证明是一种可行的解决方案。它通过学习大量未标注数据和少量已标注数据，可以在没有太多先验知识的情况下有效检测异常。从基准存储[9]来看，大多数半监督方法在仅有 1%标记异常的情况下就能超越最佳无监督方法。因此，我们使用半监督学习来训练我们的模型，不同之处在于，我们的模型在无监督学习只学习正常数据的基础上学习了一些异常数据。Transformer [10]是深度学习领域的杰出代表，它在[11]中的各种自然语言处理任务中达到了最先进水平（SOTA），并在计算机领域取得了令人印象深刻的成果。

视觉、音频和视频社区。以数据为中心的 MD 是对按时间发送的车辆信息进行检测，因此 Transformer 是我们工作的基本框架。由于车载 OBU 的计算和存储能力有限，而且在云服务器上运行检测任务存在延迟问题，因此我们采用边缘计算作为 IoV 模型的解决方案（图 1）。

在本文中，我们提出了一种基于变换器的半监督车辆错误行为检测框架（SVMD- former），该框架可将信息序列转换为错误行为分数，并通过阈值识别错误行为。本文的主要贡献如下。

- 我们设计了一个基于半监督学习和 Transformer 的新框架 SVMDformer，用于检测 IoV 中的错误行为，并在开源数据集上取得了一流的性能。
- 由于数据预处理的灵活性，我们的 SVMD- former 可以检测任何长度超过 10 的车辆序列或子序列。
- 基于半监督学习，我们的 SVMDformer 甚至可以在数据集之外检测出 19 种不当行为，并针对难以检测的不当行为类型进行有针对性的训练。
- 在实验中，我们用五个关键评估指标展示了我们的 SVMDformer，评估了它在三种监督方法和单一错误行为下的性能，并将分数可视化，分析了 EventualStop 训练量的灵敏度，比较了六个基线 NN 模型和先前的工作，并进行了消融实验。

本文接下来的内容安排如下。第二节介绍了车载网络中不当行为数据集和检测方法的相关工作。第三节介绍了 Mutil-head 自我关注的初步知识。在第四部分中，我们介绍了整个 SVMDformer 框架。此外，我们将在第四部分评估和讨论 SVMDformer 在不同环境下的性能。
V.在第六部分，我们全面分析了该模型，并将其与基线模型和基准进行了比较。最后，我们将在第七部分总结我们的工作并提出改进建议。

II. 实际工作

车辆网络入侵检测系统（IDS）是一个被广泛研究的领域。IDS 主要通过三种方式保护车辆隐私和数据安全：驾驶员身份识别、外部信息完整性检查和内部信息正确性检查。而对不当行为检测系统（MDS）的研究更侧重于检查消息的正确性，即内部人员是否发送了正确的 BSM，是一种数据驱动型 IDS。在 V2X 系统中，不当行为一般分为由 OBU、传感器和无线网络引起的故障，以及恶意攻击，包括拒绝服务（DoS）攻击、Sybil 攻击和数据回放（Data Replay）等。由于早期缺乏数据集[6]、[12]，MDS 是一个研究较晚的领域。根据最新调查[7]，有关 MDS 工作的数据集大多没有

公开，目前只有两个数据集开源。Kamel 等人分别于 2018 年和 2020 年开源了 VeReMi_ V1 [12] 和 VeReMi_ V2 [13]，这两个错误行为数据集都包含车辆发送的 BSM。VeReMi_ V1 仅包含四种位置故障和 Eventual Stop 故障。VeReMi_ V2 是 VeReMi_ V1 的改进和扩展，增加了更多字段和 19 种不当行为。在这两方面都有许多工作要做。

基于 VeReMi_ V1，[12] 的工作使用可信度检查结合 ML（SVM 和 KNN 算法）来检测位置故障，并通过使用四种检测器对五种攻击进行了比较和分析，平均 Recall 为 96.8%。Uprety 等人在论文[14]中使用联盟机器学习，在本地训练车辆数据，然后在云端平均模型，规避了车辆隐私问题，检测位置故障的准确率达到 85.71%。Ercan 等人在论文[15]中提出了一种集中训练、分布式检测的位置伪造检测算法，该算法使用了集合方法（KNN 和随机森林），平均准确率达到 85.3%。Sharma 等人在文献[16]中将可信度检查与 ML 模型相结合，并对 6 个模型的结果进行了集合投票，结果证明可信度检查是有效的，AUC 为 85.03%，准确率为 88.61%。Liu 在 [17] 中提出了一种深度学习 CNN 和 LSTM 方法来检测位置故障，准确率分别达到了 90% 和 88%。然而，上述方法大多使用传统的机器学习，不适合 V2X 系统中的大量数据。此外，VeReMi_ V1 数据集需要修订。该数据集只有五种不当行为，没有 DoS 和 Sybil 等恶意攻击，也没有加速和航向字段。

VeReMi_ V2 中的不当行为类型（从 5 种增加到 19 种）和信息字段（增加了伪 ID、加速度、航向）更多了。Kamel 等人在 [13] 中使用可信度和一致性检查检测了 19 种不当行为，准确率达到 92.93%。Mahmoudi 等人在文献[18]中从本地检测器检查中提取特征，从原始信标数据中设计附加特征，并通过 LSTM 模型检测出 19 种不当行为，最终达到了 97% 的准确率。Alladi 等人在文献[19]中提出了一种基于子序列的无监督 CNN-LSTM 架构，通过阈值算法检测出 19 种不当行为，准确率达到 98%。不过，它只能检测固定长度的信息序列。此外，同一作者的另一项研究[20]增加了更多的信息特征，并提出了三种分类方法，提高了检测效率，但只能检测到 17 种不当行为。文献[21]提出了一种基于强化学习和无分数阈值 LSTM 的检测算法，其对 19 种不当行为的检测准确率达到 98.8%。此外，还有一些研究集中于几种特定类型的不端行为，如[22]中的位置故障、[23]中的Sybil攻击和[24]中的DoS攻击。然而，现实世界中有许多不同类型的不当行为，如果只检测到其中一部分，就会增加车辆被攻击的风险。

总之，VeReMi_ V1 填补了公共数据集的空白

VeReMi_ V2 弥补了 V1 的不足。基于 VeReMi_ V2 和深度强化学习方法，检测准确率大幅提高。然而，许多作品的检测精度较低。有些作品[22]-[24]只关注少数错误行为类型。有些作品[19]、[20] 在输入方面有限制。有些作品[15]、[16]、[20] 使用多个模型进行集合或投票，这无疑会降低检测效率。因此，我们需要建立一个单一模型来有效检测所有不当行为。

III. 预备

本节将介绍有关注意力机制和缄默头自我注意的初步知识，这是我们 SVMDformer 的重要组成部分。

A. 注意机制

注意力机制是基于人类的视觉注意力，人们在观察物体时往往会将注意力集中在某些特征上。为了实现注意力机制，我们将输入视为 $< Key, Value >$ 对，并计算 *Key* 和 *Query* 之间的相似性系数，将其视为 *Value* 权重系数，然后对 *Value* 的总和进行加权，得到注意力输出。我们用 *Q*、*K*、*V* 表示 *Query*、*Key* 和 *Value*。

B. 自我关注

自我关注机制关注的是来自

内，*Q*、*K*、*V* 来自同一数据源。也就是说，*Q*、*K*、*V* 是由同一个向量通过不同的线性变换得到的。自我关注过程如下：

1. 假设输入为 $x = R^{a\times b}$ （在我们的模型中，a=200，b=128），三个线性矩阵分别为 $\boldsymbol{w}^q, \boldsymbol{w}^k \in R^{b\times d}$，以及 $w^v \in Rb\times^c$。
2. Q、K、V 变换：x 分别与三个线性矩阵相乘，得到 Q、$K \in R^{a\times d}$，以及 $V \in R^{a\times c}$：

$$\begin{cases} Q = x * \boldsymbol{w}^q \\ K = x * \boldsymbol{w}^k \\ V = x * \boldsymbol{w}^v \end{cases} \quad (1)$$

3. 乘法和比例：QK^T 和除以 $\sqrt{d_k}$ 至得到注意力得分矩阵 $G \in R^{a\times a}$，表示每个时间步之间的得分：

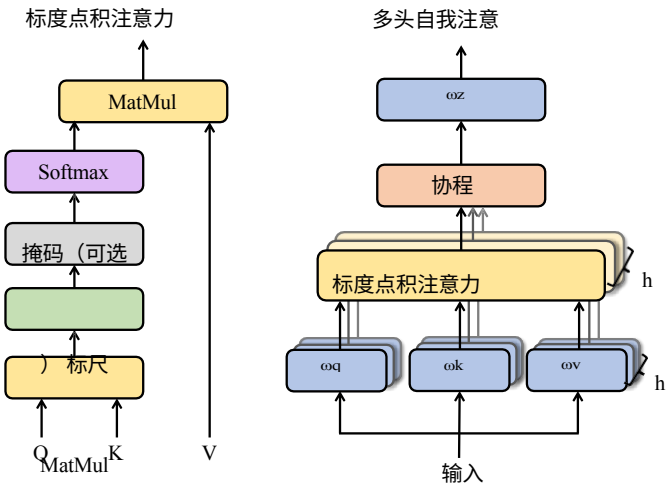


图 2：缩放点积（左）和多头自注意（右）。[10]

5. 软最大：使用 *softmax* 将分数 G 转换为注意力权重矩阵 $W \in R^{a\times a}$，表示每个时间步之间的重要性：

$$W = softmax(G) = softmax(QK^T / \sqrt{d_k}). \quad (4)$$

6. 点积：W V 产生自我注意输出矩阵 $z \in R^{a\times c}$，这是一个包含自我注意机制的注意序列：

$$\begin{aligned} z &= Attention(Q, K, V) \\ &= softmax(QK^T / \sqrt{d_k}) \cdot V。 \end{aligned} \quad (5)$$

C. 畸形头自我关注

多头自鸣器将 *Q*、*K*、*V* 成 *h* 子部分，从而使模型通过不同位置的不同表征子空间。对于单个注意头，平均化会抑制这种情况 [10]。多头自我注意有 *h* 组线性矩阵，执行步骤 2 至 6 h 次，得到 *h* 个注意输出（$z_1, ..., z_h$），将（$z_1, ..., z_h$）串联起来，并线性投影一次至获得 Mutil-head 自我关注输出：

$$MultiHead(Q, K, V) = Concat(z_1, ..., z_h)W^z \quad (6)$$

其中 $z_i = Attention(x * \boldsymbol{w}^q, x * \boldsymbol{w}^k, x * \boldsymbol{w}^v)$.

$$G = (QK^T)/\sqrt{d_k} \quad (2)$$

d_k 是 K 的维数，$1/\sqrt{d_k}$ 是防止内积值过大的缩放因子。

4. 密钥填充掩码（可选）：通过 *Kpm* 向量识别输入的填充部分：[1，1，…，0]，1 表示无填充，0 表示有填充，允许自注意层将填充部分的权重分配为 0 以忽略它：

$$G = G * Kpm^T 。 \quad (3)$$

缩放点积计算和多头自我关注机制见图 2。

IV. SVMDFORMER 框架

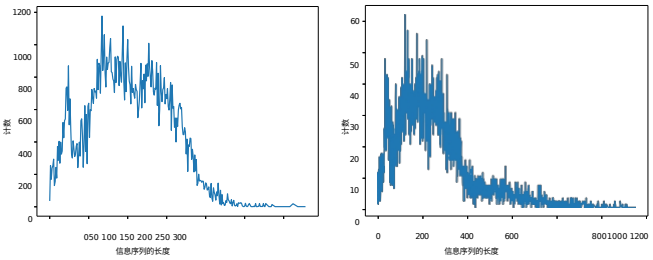
本节将介绍整个 SVMDformer 框架，并介绍 IoV 模型和不当行为模型。

A. IoV 模型

在我们设计的物联网中，RSU 部署在道路两侧，配备 SVMDformer 框架的边缘服务器部署在 RSU 中。边缘服务器通过高速有线链路与云服务器连接。

表 I：不当行为模型

身份证	类型	ID	类型
0	原创	1	常数
2	ConstPosOffset	3	随机位置
4	随机位置偏移	5	常数速度
6	ConstSpeedOffset	7	随机速度
8	随机速度偏移	9	最终停止
10	破坏性	11	数据回复
12	延迟信息	13	多达
14	随机	15	破坏性
16	网格西比尔	17	数据回复系统
18	DoSRandomSybil	19	DoSDisruptiveSybil



(a) 不含 Dos　　(b) 所含剂量

图 3：车辆信息数量统计

车辆通过无线网络与其他车辆、RSU、行人和服务器发送 BSM。在云服务器上训练好 SVMDformer 模型后，就可以在边缘服务器上运行不当行为检测。每当一辆车广播超过 10 条信息时，离其最近的 RSU 就会收到这些信息，并通过其边缘服务器检测该车是否为行为不端车辆。不轨车辆将被报告给云服务器和附近的车辆，以便用户采取行动。IoV 模型如图 1 所示。

B. 不当行为模式

不当行为包括故障和攻击。前者是由车载设备、网络或车辆传感器故障引起的非恶意行为，后者是发送错误信息的恶意行为，主要包括：故障：

1) 位置故障：车载诊断系统或传感器故障导致车辆发送错误的位置信息，包括恒定、随机、恒定偏移和随机偏移。
2) 车速故障：车载诊断系统或传感器故障导致车辆发送错误的车速信息，错误类型同上。
3) 延迟信息：网络问题导致车辆发送正确完整的信息，但与实际情况有延迟 δt。

攻击

4) DoS：攻击者发送信息的频率超过相应 IEEE 或 ETSI 标准设定的限制。
5) 数据回复：攻击者重放了之前从特定邻居收到的信息。
6) 破坏性：攻击者重放之前从随机邻居收到的信息。
7) 最终停止：攻击者模拟突然停止
冻结位置并将速度设为零。
8) GridSybil：攻击者制造虚假交通拥堵

通过伪造一个新的 ID 并保留正确的信息

见表 I。数据字段包括发送时间、发送方 ID、发送方伪 ID、信息 ID、位置、速度、加速度和航向。针对每种不当行为场景，都会生成一个地面实况（GroundTruth，GT）文件，其中包含所有车辆发送的未经修改的真实 BSM；同时，每辆车都会生成一个跟踪文件，其中包含从所有相邻车辆接收到的 BSM，其中真实车辆的信息与 GT 文件中的信息相同，而不当行为车辆会修改信息，真实车辆：不当行为车辆=7:3。

我们通过车辆的 BSM 生成训练集和测试集；一个训练/测试数据对应一辆车。没有 Dos 攻击的车辆的信息数量从 0 到 328 不等（图 3(a)），平均为 101.7 条，97.7% 以上小于 200 条。相比之下，含有 Dos 攻击的车辆由于发送间隔较短，信息量可达 1000 条（图 3(b)）或更多。为了适应神经网络的输入，我们将单个车辆的信息数量固定为 200，长的截断，短的用零补齐。在我们的实验中，Dos 包含的车辆因其独特的特征而不受截断的影响。此外，我们认为少于 10 条信息的车辆包含的信息太少，应予以舍弃。在我们的 IoV 世界中，检测从车辆进入 RSU 范围 10 秒后开始。此外，还需要遍历跟踪文件名来生成车辆的标签，0 表示 "正常"（Genuine），1 表示 "行为不端"（Misbehaving）。

我们选择了六个信息特征：发送时间、伪 ID、pos 的（x，y）、spd、acl、hed。在将 pos、spd、acl 和 hed 输入模型之前，先用 z-score 对其进行标准化处理。预处理后，车辆 i 在时间步骤 t 的原始序列为

$$Seq_t^i = \{\Delta T_t^i,\ SumP^i,\ pos_{tx}^i,\ pos_{tx}^i,\ spd_{tx}^i,\ spd^i,\ acl^i,\ acl^i,\ hed^i,\ hed^i\} \quad (7)$$

、 tx tx ty tx ty

t

频率

VeReMi_ V2 中包含的 19 种不当行为如图所示表一

C. 数据集和预处理

我们使用 VeReMi_ V2 [13] 数据集来评估模型性能，该数据集包含 19 种不当行为类型，如图所示

其中 ΔT^i 表示信息发送间隔：

$$\Delta T_t^i = \text{发送时间}_t - \text{发送时间}_{t-1}, \tag{8}$$

$^i i,\ t \in \mathbb{N}^+,\ t \in [0,\ 199]$。因此，原始序列（$Seq^i$）的大小为 200 × 10。

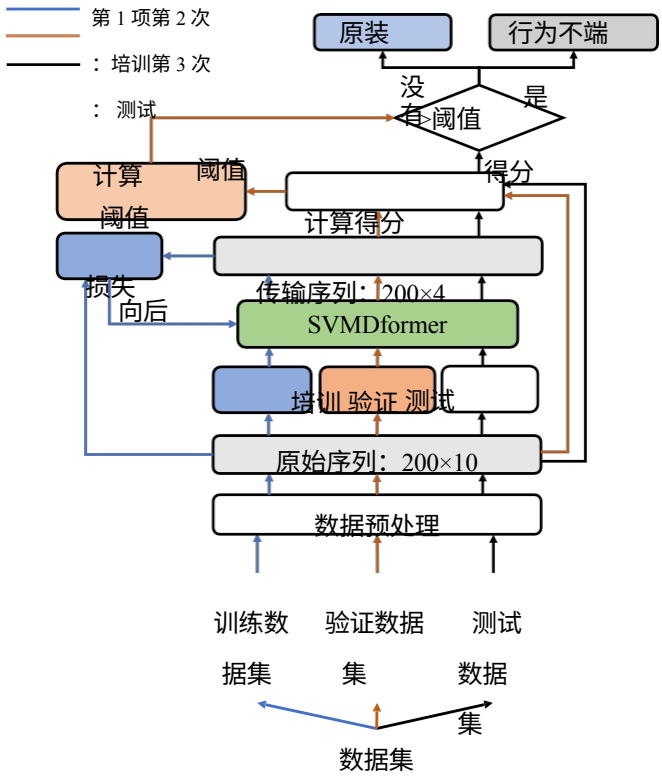


图 4：SVMDformer 框架。

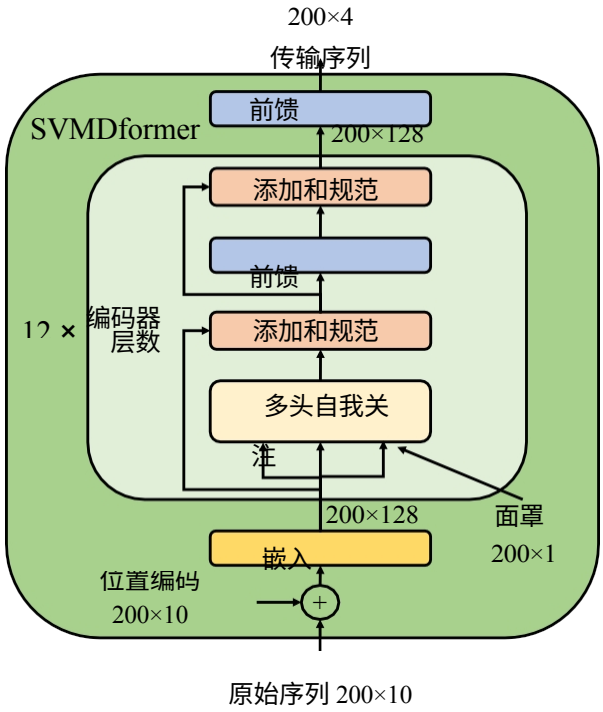


图 5：SVMDformer 模型架构。

D. SVMDformer 框架

我们工作中提出的 SVMDformer 框架（图 4）分为三个阶段：训练、验证和测试。训练和测试是交替进行的，模型将在收敛后的每个纪元进行训练和验证，我们在训练过程中将模型的最佳纪元保存在验证集上。

1) 第 1 阶段：训练，在训练集上使用损失反向传播（反向）算法优化模型参数。
2) 第二阶段：验证，用训练有素的模型计算验证集数据的分数，并根据分数和标签计算 AUC 和错误行为分类阈值。
3) 第三阶段：测试，衡量模型的最终性能。用分数计算 AUC；用第 2 阶段的阈值计算测试集的准确度、F1、召回率和精确度。

经过数据预处理后，原始序列被输入模型，并转化为包含位置和速度特征的反序列。然后，我们计算两个序列的相似度，在训练中构建损失函数，在验证和测试中表示错误行为得分。在训练过程中，模型会学习大量真实数据和少量难以识别的行为不端数据。包括 EventualStop 在内的错误行为车辆势必不符合模型学习到的数据分布特征，从而导致反序列偏离原始序列。因此，在验证和测试过程中，行为不端车辆的行为不端得分也会高于未标记车辆和真实车辆，这样就可以为行为不端分类设定一个阈值。

E. SVMDformer 模型架构

我们工作中提出的 SVMDformer 是在变换编码器的基础上进行微调的。SVMDformer 模型将 200×10 个原始序列转换为 200×4 个变换序列。模型结构如图 5 所示。SVMDformer 由包含位置编码层和嵌入层的输入层、包含 12 个相同层的编码器和输出层组成。我们保留了 Transformer 的普通编码器层的结构，并对输入层和输出层进行了微调。每个编码器层有三层：多头自注意（Attention）、Add&Norm（ResNet [25]）和前馈（FF）。SVMDformer 的每一层如下：

1) 位置编码：变换器内部由线性层组成，线性层是没有序列信息的并行计算结构。为了提取输入的序列特征，我们在原始序列中加入香草 sin/cos 编码[10]，计算

公式为

数据（EventualStop）；训练目标是使

未标记车辆和真实车辆的反序列符合其原始序列，而行为不端车辆偏离了

原始序列。这样一来，由于模型已经学会了许多真正的行为模式，所以行为不端的车辆也就不存在了、

(

$$PE_{(pos,2i)} = sin(\text{pos}/10000^{2i/d_{msg}}) \quad (9)$$

、

$$PE_{(pos,2i+1)} = cos(\text{pos}/10000^{2i/d_{msg}})$$、

其中，i 是位置，pos 是位置 i 上的值，d_{msg} 是原始序列的维度。之后

将 200 10 个位置编码与 200 10 个原始序列相加，就得到了 200 10 个位置序列。

2) 嵌入层：我们在编码器之前添加了一个嵌入层，目的是将位置序列扩展到
200 128 个嵌入序列。更多的神经元会加强编码器层，学习序列的数据分布。
3) 注意层：我们将嵌入序列（或第一个编码器层之后的编码器序列）输入 Mutil-head 自注意层，并输出 200 128 注意序列。有关 Mutil-head 自我注意的详情，请参阅第 III-B 和 III-C 节。
4) 前馈层：注意层之后是前馈层
层由两个线性变换组成，中间有一个 Relu 激活，以增加其非线性。层将注意力序列转换为 200 128 前馈序列。
5) ResNet：表示为 *LayerNorm*(*x* + *Sublayer*(*x*))。在编码器中，*子层*是多头自注意（Multi-head self-attention）和前馈（FeedForward），不改变输入数据，这是执行添加操作的关键。ResNet 可以保留 *Sublayer* 的输入和输出数据特征，有效避免了深度网络带来的遗忘问题。添加残差后，使用 LayerNorm 进行归一化处理，以避免梯度消失和爆炸问题。
6) 输出层：在 12 层编码器之后，我们增加了一个前馈层，将 200 128 编码器序列转换为 200 4 转码序列。这样，我们就可以比较转码序列和原始序列之间的位置和速度差异。

F. 损失函数

我们使用损失反向传播（backward）算法来优化模型参数，因此需要设定一个训练目标（最小*损失*）。首先，我们使用平均绝对误差
(*MAE*) 来表示序列之间的相似性。车辆的主要故障是位置和速度，我们仅通过比较位置和速度特征来获得最佳检测性能。因此，反向序列只有四个特征指示
计算位置（x，y）和速度（x，y），而原始序列只需提取位置和速度字段，因此：

其中，*n* 是未标记车辆的数量，*m* 是已标记车辆的数量，y_j 是车辆的半监督标记，真实为 1，行为不端为-1。λ 是已标记车辆的折扣系数，如果 $\lambda > 1$，则强调已标记车辆的重要性；如果 $\lambda < 1$，则强调未标记车辆的重要性。

由于 VeReMi 数据集中的车辆标签都是已知的，我们将一些真实车辆视为未标签车辆，以区分已标签车辆和未标签车辆，并突出 λ 的作用。然而，我们在实验中发现，当 $\lambda = 1$ 时，模型的结果最佳，因此当 $\lambda = 1$ 时，*损失*函数为：

$$损失 = \frac{1}{n*m}\left(\sum_{n} MAE + \sum_{m} \frac{1}{MAE_j}\right), \quad (12)$$

其中，*n* 为真正车辆的数量，*m* 为行为不端车辆的数量。

G. 分数和阈值

在验证和测试阶段，我们不计算*损失*，而只计算原始序列和反序列之间的 *MAE*，即行为失常*得分* (*S*)，简称为 "*S*"：

$$S = MAE = \frac{1}{l*w}\sum |y - \hat{y}|, \quad (13)$$

然后，我们计算 AUC 来评估模型的检测效率。AUC（Area Under the roc Curve）是异常检测中的一个重要指标。假设有 *n* 辆正常（*G*）车辆和 *m* 辆行为不端（*M*）车辆，因此有 *n m* 对相反的样本。每对样本的不轨行为得分用 $< S_G , S_M >$ 表示，AUC 等于这些样本对中 $S_G > S_M$ 的概率，公式如下：

$$AUC = \frac{1}{n*m}\sum f(S_G, S_M),$$

$$\text{其中} f(S_G, S_M) = \begin{cases} 1, & \text{如果 } S_G > S_M \\ 0.5, & \text{如果 } S_G = S_M \\ 0, & \text{如果 } S_G < S_M \end{cases} \quad (14)$$

然后，我们使用迭代算法（算法 1）找到最佳分类阈值

$$MAE = \frac{1}{l * w} \sum_{i}^{l} \sum_{j}^{w} |y_i^j - \hat{y}_i^j|, \quad (10)$$

其中，$l = 200$，$w = 4$，y_i^j 和 $\hat{y}_i^j$ 表示原始序列和反序列中的值。

采用半监督方法拟合反式序列和未标记车辆与真实车辆的原始序列，而少量标记的行为不端车辆则远离原始序列，*损失*函数为：

$$\text{损失} = \frac{1}{n * m} \sum_{i}^{n} MAE + \frac{\lambda}{i} \sum_{n * m}^{m} (MAE)^{y_j}, \quad (11)$$

，对车辆进行分类。
行为的验证集。由于行为不当的 S 值较高

车辆，而低价为正品时，该车辆被视为
如果 $S > Threshold$，则为行为不端；反之，则为真实。

最后，我们计算**准确率、F1-分数、召回率、精确**率**和 AUC（公式** (14)**）**，以评估模型在测试和验证阶段的检测性能。

V. 模型性能评估

本节评估了模型的检测性能，包括三种监督下的检测、单一错误行为的检测以及得分可视化
的单一不当行为。

算法 1：阈值算法

输入：*分数、标签、纪元*

输出：*阈值*

精确度 ← 0.1

阈值 ← 1

次迭代 ← 18

foreach *Epochs* **do**

  精度 ← [0]*迭代*

  threshold ← [0] ∗ *iteration*

  foreach *i in iteration* **do**

    阈值[*i*] ← *最佳阈值* + *精确度* ∗(*i* − *迭代*/2)

    用*阈值*[*i*]计算*精确度*[*i*]

  结束

  maxAcc_ idx ← argmax(*准确度*)

  阈值 ← *阈值*[*maxAcc_ idx*]

  精确度 ← *精确度* ∗ 0.1

结束

返回

表 II：SVMDformer 的超参数

名称	价值	说明
纪元	160	模型迭代次数
种子	1	随机种子
学习率	1e-4	OneCycle [26] 策略，最小 lr= 1e-4
启动	Relu	前馈层的激活
辍学率	0.0	神经元丢失率
优化器	AdamW	亚当+体重十年 [27]
层数	12	编码器层数
n_es	250	EventualStop 训练量的数量
批量大小	256	小批量样品数量
d 序列	200	源序列长度
d_msg	10	信息特征尺寸
_d	128	嵌入序列的维度
d_ff	512	前馈
_d	4	跨序列尺寸
h	4	自我关注头数
等	1	错误数据丢失的因素

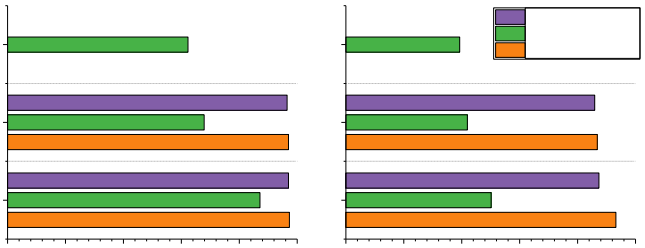
A. 实验环境

实验基于深度学习平台 Pytorch 1.11.0、Python 3.8 和 Cuda 11.3；硬件环境依赖于云服务器、显卡：RTX A5000 * 1，内存：24GB。CPU：Intel(R) Xeon(R) Platinum 8358P，内存：80GB。

将训练集、验证集和测试集随机分为三个独立子集及其

表 III：三种监督下的检测方法

方法	列车组	损失	AUC	Acc	F1	Rec	前
无监督	62295:0	MSE	99.83	99.30	99.29	99.30	99.36
		MAE	99.85	99.37	99.36	99.37	99.45
监督	62295:500*19	CE	98.12	96.96	96.81	96.96	97.72
		$MSE^{y^{'}}$	98.39	97.10	96.97	97.10	97.71
		$MAE^{y^{'}}$	99.36	97.51	97.38	97.51	98.06
半监督	62295:250	$MSE^{y^{'}}$	99.85	99.34	99.34	99.34	99.36
		$MAE^{y^{'}}$	**99.87**	**99.66**	**99.66**	**99.66**	**99.68**



(a) AUC (b) Acc

真实数据：错误行为数据 = 62295：500*19、5700：300*19 和 24700：1300*19，其中 19 种错误行为被视为等量的真实数据。在训练集中，真实数据将被完全学习，而行为不当数据则取决于实验。SVMDformer 的主要超参数如表 II 所示。

我们使用 AdamW [27] 作为模型的优化器，它在 Adam 的基础上增加了权重衰减，是目前最好的优化器之一。学习率调整策略

图 6：每种监督方法的 AUC 和准确率。

是 Onecycle [26]：学习率首先从最初的_ lr=1e-7 升温至最大_ lr=2e-3 (10 epochs)，然后降至最小_ lr=1e-4 (40 epochs)，并在最后 110 epochs 内缓慢降至 2e-6。辍学率为 0.0，因为过度拟合有利于我们的模型，其目的是使未学习到的行为不端车辆偏离模型。

B. 三类监督下的检测

本小节通过学习不同数量的行为不端车辆来比较三种监督方法。训练集和结果如表 III 所示，$\tilde{y}$ 是损失函数中的伪标签，正品为 1，不守规矩为-1。在监督学习中，我们还使用交叉熵（CE）损失函数对二元分类方法进行了比较。模型的输出不再是一个反序列，而是一个 1 2 向量，代表真实和不当行为的概率。然而，这种方法无法反映与原始序列的差异，在降尺度过程中，表示数据分布的能力也会降低。从表 III 中可以看出，它的性能最差。

如表 III 和图 6 所示，在三种超视觉方法中，使用 MAE 的相似度函数优于 MSE。与 MAE 相比，无监督模型仅学习了 62 295 辆真实车辆，效果良好，AUC 达到 99.85%，准确率为 99.37%。有监督模型对 19 种不当行为中的每一种都额外学习了 500 辆车，但所有指标都是最低的。半监督模型的准确率最高，AUC 为 99.87%，准确率为 99.66%，但它只学习了 **250 辆 EventualStop 车辆**。比较图 6 中的 AUC 和准确率，可以明显看出，半监督方法比无监督方法略好，比有监督方法好得多。

C. 检测单一不当行为

每次检测单一不端行为时，测试集中只有一种类型的不端行为，其中真实数据：不端行为数据=1300:1300。表 IV 客观显示了模型对每种不当行为的检测性能，其中 Genuine 结果是 19 次模拟中 Genuine 类的加权平均值，MixAll 结果是 20 种行为的加权平均值。

可以看出，在无监督学习的情况下，除了 EventualStop（ES）类型之外，模型都取得了不错的结果。ES 类型的召回率仅为 80.7%，反映出模型遗漏了大量此类不当行为，这表明 EventualStop 和 Genuine 之间的高度相似性导致模型将 ES 误认为 Genuine。半监督模型只额外学习了 250 辆 ES 类型的不当行为车辆，EventualStop 的召回率提高到 92.34%。ConstPos 和 RandomPos 等 14 种错误行为的召回率均高达 100%。模型对这些不当行为没有遗漏，MixAll 的准确率高达 99.66%，可以说是一个近乎完美的检测器。

相比之下，监督模型的结果在几乎所有错误行为环境中都有所下降，尤其是随机- PosOffset。在学习了 19 种错误行为后，模型开始向奇怪的反向学习。RandomPosOffset 的检测结果变得难以预测，召回率仅为 42.95%。这表明，由于不当行为类型的多样性，监督学习并不是不当行为检测的好选择。

D. 分数可视化

此外，我们还通过 Score 散点图和 Threshold 线讨论了 SVMDformer 在单一不当行为场景中的表现，如图 7 所示。为便于展示，我们选取了正版类中排名前 150 位的数据和行为不端类中排名最低的 150 位数据，即每次检测中阈值附近的 300 个数据。

总体而言（图 7(a)），类型 3、7、12、14、18 和 19 的平均得分都超过了 100 分，甚至 1000 分，这是因为它们具有明显的行为失常特征。这与平均得分低于 0.004 的真实数据差距很大，其中也有极少量的异常值。在约 0.02 的阈值附近（图 7(b)），虽然有少量行为不端数据被错误识别，但大部分数据都在阈值以上。然而，类型 9（EventualStop）有大量低于阈值的出现（实际上为 7.66%*1300=100），模型没有识别出来，导致该类型的召回率仅为 92.34%（表四）。此外，可以看到类型 2（ConstPosOffset）非常接近阈值，这也是一种类似 Genuine 的错误行为。

VI. 分析与讨论

在本节中，我们分析了 EventualStop (ES) 训练量的灵敏度，将 SVMDformer 与六个基线模型和先前的工作进行了比较，并进行了消融实验。

A. 对 EventualStop 的敏感性分析

我们将真实数据的数量固定为 62 295 个，并以 50 个为增量，探索 SVMDformer 学习 0 到 500 个 EventualStop 数据的效果。

从表 V 和图 8（b）可以看出，随着 ES 训练量的增加，ConstPosOffset、RandomPosOff-set、DataReply 和 Dos 的召回率有所波动，但与 ES 没有相关性。从图 8(a)可以看出，ES 的召回率随着训练量的增加而增加，这表明训练量越大，识别率越高。然而，一些真正的车辆在最后几步移动缓慢，这与 ES 的错误行为十分相似。因此，模型会将一些真实数据错误地分类为 ES 类型，导致 Genuine 结果下降（图 8(b) 中的粉红色曲线）。然而，当训练量超过 250 时，EventualStop 的上升趋势和 Genuine 的下降趋势开始放缓。模型在 250 时达到最佳结果，MixAll 的召回率为 99.66%。

总之，当 EventualStop 的训练量为 250 时，真正车辆和违规车辆之间的检测是平衡的，结果达到最佳。

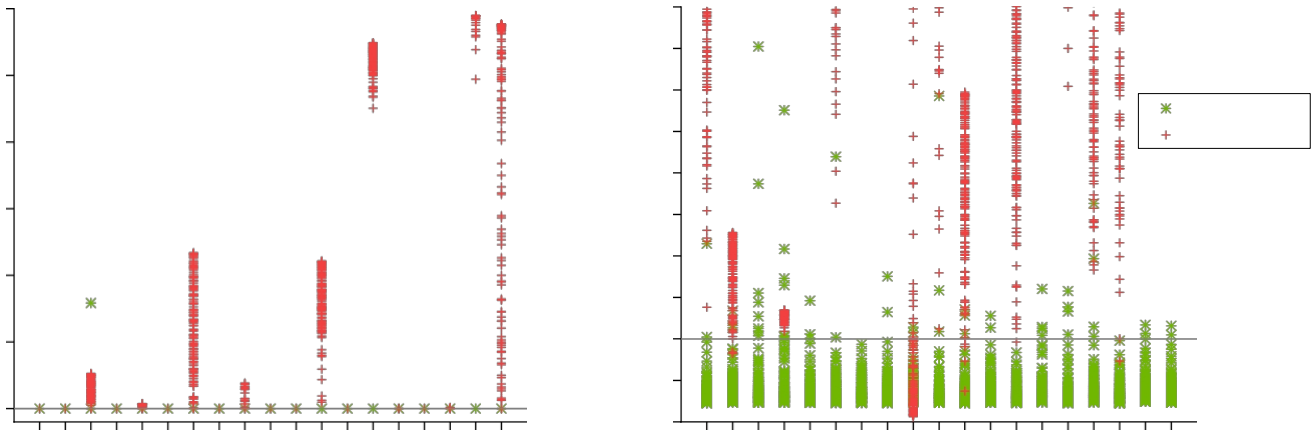
B. 神经网络模型比较

本小节通过在不改变 SVMDformer 框架（图 4）其他模块的情况下更换模型，对六种基线神经网络模型进行了比较。各模型和结果（见表 VI）如下。

1) CNN：2 × (Convolution + Batchnorm + Relu activation)，然后进行最大池化。从表 VI 中可以看出，CNN 在时间序列数据方面表现不佳，AUC 仅为 97.77%，准确率仅为 94.35%。
2) Bi-LSTM：Bi-LSTM 有两个 512 维的隐藏层、LayerNorm 和前馈层。其 AUC 与 SVMDformer 一样高，达到 99.87%，证实了 LSTM 在时间序列方面的优势。但是，其他指标还不够理想。
3) CNN-BiLSTM：使用模型 1 提取局部特征，然后使用模型 2 提取时间特征，求三个特征向量的平均值。AUC 99.92% 属于最佳结果，但准确率为 99.1%，低于 BiLSTM。
4) ResNet-18 [25]：四个残差块，每个块有四个一维卷积层，加上第一个卷积层和最后一个线性层，共 18 层。其 AUC 为 99.1%，远高于 CNN，证明了其优越性。
5) 前馈：双层线性，添加位置编码，中间维度为 512。AUC 仅为 93.94%，准确率仅为 89.23%。由于两层线性过于简单，即使添加了位置编码，结果也很糟糕。
6) 请注意用单个 Mutil-head 自注意层取代 12 层编码器。而 AUC 和 Acc 分别比 CNN 和 ResNet 高出 97.55% 和 94.97%，证明了 Mutil-head 自注意力的优越性。

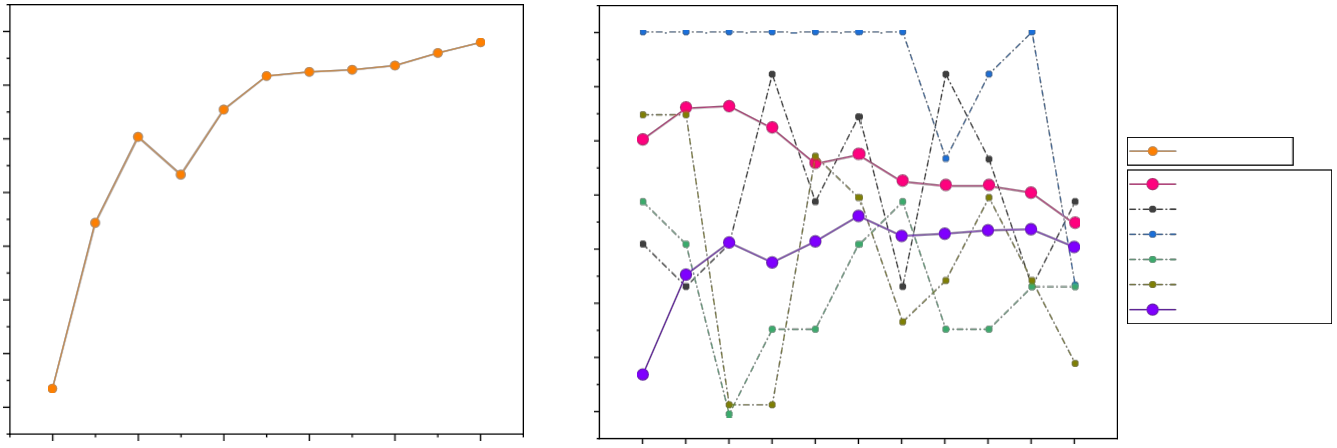
表 IV：单一不当行为的检测

		半监督					无人监督					监督				
身份证	类型	AUC	Acc	F1	Rec	前	AUC	Acc	F1	Rec	前	AUC	Acc	F1	Rec	前
0	原装	99.87	99.66	99.67	99.77	99.58	99.85	99.37	99.41	99.80	99.10	99.36	97.51	97.87	99.14	97.05
1	常数	100.00	99.88	99.88	100.00	99.77	100.00	99.96	99.96	100.00	99.92	99.95	99.49	99.49	99.84	99.14
2	ConstPosOffset	100.00	99.84	99.84	99.84	99.84	100.00	99.69	99.69	99.61	99.76	99.57	92.59	92.08	86.13	98.92
3	随机位置	99.90	99.57	99.57	100.00	99.15	100.00	99.77	99.77	100.00	99.53	100.00	99.41	99.42	100.00	98.84
4	随机位置偏移	99.86	99.77	99.77	100.00	99.53	99.93	99.77	99.77	100.00	99.53	90.58	71.01	59.70	42.95	97.87
5	常数速度	100.00	99.92	99.92	100.00	99.84	100.00	99.88	99.88	100.00	99.76	100.00	99.49	99.49	100.00	98.99
6	常数速度偏移	100.00	99.92	99.92	100.00	99.84	100.00	99.88	99.88	100.00	99.77	100.00	99.76	99.77	100.00	99.53
7	随机速度	100.00	99.96	99.96	100.00	99.92	100.00	100.00	100.00	100.00	100.00	100.00	99.45	99.45	100.00	98.91
8	随机速度偏移	100.00	99.88	99.88	100.00	99.77	100.00	99.96	99.96	100.00	99.92	100.00	99.65	99.65	100.00	99.30
9	最终停止	97.72	96.09	95.94	92.34	99.83	97.22	90.27	89.24	80.70	99.81	97.84	96.64	96.57	94.45	98.77
10	破坏性	99.95	99.81	99.81	100.00	99.61	100.00	99.81	99.81	99.92	99.69	99.97	99.57	99.57	99.84	99.30
11	数据回复	100.00	99.76	99.76	99.61	99.92	100.00	99.76	99.76	99.69	99.84	99.87	98.78	98.78	98.35	99.21
12	延迟信息	100.00	99.92	99.92	100.00	99.84	100.00	99.96	99.96	100.00	99.92	100.00	99.37	99.38	100.00	98.76
13	多斯	100.00	99.77	99.77	99.69	99.85	100.00	99.89	99.89	99.85	99.92	100.00	99.62	99.62	100.00	99.24
14	随机	100.00	99.81	99.81	100.00	99.62	100.00	99.85	99.85	100.00	99.69	100.00	99.58	99.58	100.00	99.16
15	DosDisruptive	100.00	99.89	99.89	100.00	99.77	100.00	99.89	99.89	100.00	99.77	100.00	99.69	99.69	100.00	99.39
16	网格西比尔	100.00	99.92	99.92	100.00	99.85	100.00	99.96	99.96	100.00	99.92	100.00	99.43	99.43	100.00	98.86
17	数据回复西比尔	100.00	99.96	99.96	99.92	100.00	100.00	99.96	99.96	99.92	100.00	99.97	99.61	99.61	99.92	99.30
18	DosRandomSybil	100.00	99.96	99.96	100.00	99.92	100.00	99.88	99.88	100.00	99.77	100.00	99.73	99.73	100.00	99.46
19	DosDisruptiveSybil	100.00	99.92	99.92	100.00	99.85	100.00	99.85	99.85	100.00	99.69	100.00	99.65	99.65	100.00	99.31
20	混合所有	**99.87**	**99.66**	**99.66**	**99.66**	**99.68**	99.85	99.37	99.36	99.37	99.45	99.36	97.51	97.38	97.51	98.06



(a) 总体情况　　(b) 临近门槛

图 7：单一不当行为场景的阈值和分数。



(a) 最终停止　　(b) 五种行为和 MixAll

图 8：不同 ES 训练量下六种行为和 MixAll 的召回率。

表五：不同 ES 训练量下的召回率

类型	0	50	100	150	200	250	300	350	400	450	500
0	99.80	99.86	99.86	99.82	99.76	99.77	99.73	99.72	99.72	99.71	99.65
1	100	100	100	100	100	100	100	100	100	100	100
2	99.61	99.53	99.61	99.92	99.69	99.84	99.53	99.92	99.76	99.53	99.69
3	100	100	100	100	100	100	100	100	100	100	100
4	100	100	100	100	100	100	100	99.77	99.92	100	99.53
5	100	100	100	100	100	100	100	100	100	100	100
6	100	100	100	100	100	100	100	100	100	100	100
7	100	100	100	100	100	100	100	100	100	100	100
8	100	100	100	100	100	100	100	100	100	100	100
9	80.70	86.88	90.08	88.67	91.09	92.34	92.50	92.58	92.73	93.20	93.59
10	99.92	99.84	99.84	99.92	100	100	99.92	100	99.92	100	100
11	99.69	99.61	99.29	99.45	99.45	99.61	99.69	99.45	99.45	99.53	99.53
12	100	100	100	100	100	100	100	100	100	100	100
13	99.85	99.85	99.31	99.31	99.77	99.69	99.46	99.54	99.69	99.54	99.39
14	100	100	100	100	100	100	100	100	100	100	100
15	100	100	100	100	100	100	100	100	100	100	100
16	100	100	99.85	100	100	100	99.92	100	100	100	100
17	99.92	99.92	99.84	99.92	99.92	99.92	99.92	100	100	100	99.92
18	100	100	100	100	100	100	100	100	100	100	100
19	100	100	100	100	100	100	100	100	100	100	100
20	99.37	99.55	99.61	99.58	99.61	99.66	99.63	99.63	99.64	99.64	99.60

表 VI：NN 模型比较

型号	级别	F1	Rec	前	AUC	Acc
SVMDformer	真正的行为 不端	**99.67** **99.65**	**99.77** **99.55**	**99.58** **99.77**	99.87	**99.66**
CNN	真正的行为 不端	94.90 93.50	96.20 92.50	94.32 95.89	97.77	94.35
BiLSTM	真正的行为 不端	99.33 99.31	99.42 99.22	99.26 99.42	99.87	99.32
CNN-BiLSTM	真正的行为 不端	99.12 99.12	98.71 99.52	99.54 98.72	**99.92**	99.12
ResNet	真正的行为 不端	95.16 93.49	95.95 93.14	95.12 95.51	99.11	94.54
前馈	原装 行为不端	90.91 86.45	97.16 81.30	86.61 96.21	93.94	89.23
请注意	真正的行为 不端	95.84 93.35	98.66 91.27	94.04 98.40	97.55	94.97

总之，ResNet 优于无残差的 CNN，Attention 层简单但有效，时态模型 LSTM 在 AUC 上表现良好，但其他指标均不如我们的 SVMDformer，这证明了 SVMDformer 的稳定性和鲁棒性。

C. 消融

在不改变编码器层基本结构的情况下，我们删除了拟议 SVMDformer 上的每个模块，以判断每个模块的重要性。SVMDformer 的主要模块包括位置编码（PE）和嵌入编码。

表 VII：消融

型号	级别	F1	Rec	前	AUC	Acc

表八：SVMDformer 和以前的工作

检测模型	AUC	Acc	前	Rec	F1
SVMDformer	**99.87**	**99.66**	**99.68**	99.66	**99.66**
RL & LSTM [21]	-	98.82	97.24	**99.70**	98.45
DeepADV [19]	-	98.0	99.60	95.60	97.60
DLCE [20]	-	99.63	96.86	96.74	96.75
	-	97.09	95.81	95.59	95.58
	-	98.24	96.57	96.49	96.49
VeReMi V2 [13]	-	92.93	99.12	82.28	89.92
LSTM [18]	-	97	94.5	91.77	93

注：DLCE 只检测到 17 种不当行为。其他作品为 19 种。

输入层中的 "剔除"（Emb）、注意层中的 "键-填充-掩码"（Kpm）和输出层中的 "前馈"（FF）。反过来，我们将这四个模块从 SVMDformer 中移除，其中 FF 层由线性层取代，用于转换为反式序列。表 VII 中的结果显示

- 在没有 PE 的情况下，AUC 和准确率分别下降了 0.5% 和 2.8%，这证明了位置信息对于并行计算线性变换的重要性，而位置信息正是我们 SVMDformer 的组成部分；
- 如果不使用 Emb，AUC 会降低 1.1%，Ac-..... 这表明，将教育部门的工作范围扩大到所有教育机构，可使这些机构的工作时间缩短 3.6%。数据的特征维度可以大大提高模型的学习能力；
- 如果没有 Kpm，影响是很小的，这意味着让注意层学习空信息并不意味着对正常信息的学习产生显著影响；
- 在不使用 FF 的情况下，用线性变换替换对模型的影响也很小，只是略微降低了其缩放时的特征提取能力

原装 **99.67** **99.77** **99.58**

总之，SVMDformer 的每个模块都对模型产生了积极影响，尤其是位置编码和嵌入层。

D. 与以往工作的比较

最近，VeReMi_ V2 上出现了许多关于不当行为检测的作品（表八）。

Alladi 等人在DeepADV[19] 中对19 种错误行为的检测

SVMDformer	行为不端	**99.65**	**99.55**	**99.77**	**99.87**	**99.66**
无 PE 无 Emb	原装	97.28	99.21	95.93	99.36	96.85
	行为不端	96.20	94.50	99.13		
	原装	96.67	98.75	95.30	98.77	96.08
	行为不端	95.00	93.41	98.56		
无 Kpm	原装	99.54	99.72	99.37	99.85	99.53
	行为不端	99.52	99.34	99.72		
没有 FF	原装	99.59	99.65	99.55	99.86	99.59
	行为不端	99.58	99.53	99.65		

准确率达到98%，但他们在输入（固定20 10）上有限制。此外，他们在 DLCE[20] 中也在此基础上进行了改进，检测性能有所提高，但他们取消了对 EventualStop 和 DelayedMessages 的检测，并提出将标签作为车辆特征进行训练。与 SOTA 的工作[21] 相比，除了召回率略低外，我们的工作在其他指标上都有很大改进。AUC 为 99.87%，而

而其他作品则缺乏这一重要指标。精确度和 F1 分数分别提高到 99.68% 和 99.66%，特别是检测精度从 98.82% 提高到 99.66%。误检率从 1.18% 降至 0.34%，降低了三倍多。

VII. 结论

在这项工作中，我们提出了一个以数据为中心的车辆错误行为检测框架 SVMDformer，该框架基于半

监督学习和变换器。我们在云服务器中训练 SVMD 前模型，并在边缘服务器中对物联网中的车辆 BSM 执行不当行为检测。在我们的框架中，我们将车辆信息序列转换为不当行为得分，并将不当行为分类为一个阈值。在实验中，我们全面展示了模型的检测性能。我们比较了三种监督方法，发现半监督的效果最好。我们通过可视化分数对每种不当行为进行单独检测和讨论。我们以难以识别的类型（EventualStop）为学习目标，并找到了最佳的 250 ES 训练量。我们比较了基线模型，证明了 SVMDformer 的优越性。我们对 SVMDformer 进行了消融实验，证明所有模块都有积极效果，尤其是位置编码和嵌入模块。与 SOTA 工作[21]相比，我们在 AUC、Accuracy、Precision 和 F1 指标上表现出色，误检率降低了三倍多！此外，我们提出的 SVMDformer 框架可以检测到 19 个超过 10 个车辆序列的不当行为。我们还能基于半监督检测未学习到的或超出数据集的不当行为。因此，我们的 SVMDformer 是目前最通用、最准确、最全面的不当行为检测框架。

然而，SVMDformer 只能识别车辆是否存在不当行为，而不能识别其具体类型。因此，今后我们将专门讨论如何识别确切的不当行为类型，并将 SVMDformer 框架有效地部署到物联网仿真系统中，用于不当行为检测。

参考资料

[1] 彭博社。2010 年至 2022 年全球汽车销量，2023 年预测。(2022 年 10 月 25 日）。

[2] Rim Gasmi 和 Makhlouf Aliouat.车载特设网络与车联网--比较观点。*2019年网络与先进系统国际会议（ICNAS）*，第1-6页，2019年。

[3] Yushan Siriwardhana、Pawani Porambage、Madhusanka Liyanage 和 Mika Ylianttila。使用 5G 移动边缘计算的移动增强现实调查：架构、应用和技术方面。*IEEE Communications Surveys Tutorials*，23（2）：1160-1192，2021.

[4] Seng W. Loke.合作式自动驾驶汽车：超越联网的社会智能车辆的机遇与挑战综述。*IEEE Transactions on Intelligent Vehicles*, 4(4):509-518, 2019.

[5] Salabat Khan, Fei Luo, Zijian Zhang, Mussadiq Abdul Rahim, Mubashir Ahmad, and Kaishun Wu. 车载公钥基础设施（Vpki）问题与最新进展调查。*IEEE Communications Surveys Tutorials*，24（3）：1574-1601，2022.

[6] Rens Wouter van der Heijden、Stefan Dietzel、Tim Leinmu¨ller 和 Frank Kargl. 合作式智能交通系统中的不当行为检测调查。*IEEE Communications Surveys Tutorials*, 21(1):779-811, 2019.

[7] Abdelwahab Boualouache 和 Thomas Engel.基于机器学习的 5G 及以上车载网络不当行为检测系统调查。*IEEE Communications Surveys Tutorials*，第 1-1 页，2023 年。

[8] Lukas Ruff, Robert A Vandermeulen, Nico Go¨rnitz, Alexander Binder, Emmanuel Mu¨ller, Klaus-Robert Mu¨ller, and Marius Kloft.Deep semi-supervised anomaly detection. *arXiv preprint arXiv:1906.02694*, 2019.

[9] 韩松桥、胡西洋、黄海亮、蒋明琪和赵越。Adbench：异常检测基准。*神经信息处理系统（NeurIPS）*。

[10] Ashish Vaswani、Noam Shazeer、Niki Parmar、Jakob Uszkoreit、Llion Jones、Aidan N Gomez、Łukasz Kaiser 和 Illia Polosukhin。注意力就是你所需要的一切。*神经信息处理系统进展*》，2017年30期。

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova.Bert：用于语言理解的深度双向变换器的预训练。*ArXiv preprint arXiv:1810.04805*, 2018.

[12] Rens W van der Heijden、Thomas Lukaseder 和 Frank Kargl。Veremi：用于对虚网中不当行为检测进行可比评估的数据集。*通信系统安全与隐私国际会议*，第 318-337 页。Springer, 2018.

[13] 约瑟夫-卡迈勒、迈克尔-沃尔夫、伦斯-W-范德黑、阿尔诺-凯泽、帕斯卡尔-乌里安和弗兰克-卡格尔。Veremi 扩展：用于比较评估虚网中不当行为检测的数据集。在 *ICC 2020 - 2020 年电气和电子工程师学会国际通信会议（ICC）*，第 1-6 页，2020 年。

[14] Aashma Uprety、Danda B. Rawat 和 Jiang Li.使用联合机器学习的 iov 中隐私保护不当行为检测。*2021 年电气和电子工程师学会第 18 届消费者通信网络年会（CCNC）*，第 1-6 页。

[15] Secil Ercan、Marwane Ayaida 和 Nadhir Messai。利用机器学习检测虚网中位置伪造攻击的不当行为。*IEEE Access*，10：1893-1904，2022。

[16] Prinkle Sharma 和 Hong Liu.基于机器学习、以数据为中心的车联网不当行为检测模型。*IEEE 物联网期刊*，8（6）：4991-4999，2020.

[17] Xiangyu Liu.基于深度学习的网络不当行为检测。*2022 年网络、通信与信息技术国际会议（CNCIT）*，第 122-128 页。IEEE, 2022.

[18] Issam Mahmoudi、Joseph Kamel、Ines Ben-Jemaa、Arnaud Kaiser 和 Pascal Urien。在 C-its 中实现可靠的基于机器学习的全局不当行为检测：模型评估方法。*智能城市的车载 Ad-hoc 网络*》，第 73-86 页。Springer, 2020.

[19] Tejasvi Alladi、Bhavya Gera、Ayush Agrawal、Vinay Chamola 和 Fei Richard Yu。Deepadv：用于车载异常检测的深度神经网络框架。*IEEE Transactions on Vehicular Tech- nology*，70（11）：12013-12023，2021.

[20] Tejasvi Alladi、Varun Kohli、Vinay Chamola 和 F Richard Yu。合作式智能交通系统入侵检测中基于深度学习的不当行为分类方案》。*数字通信与网络*，2022 年。

[21] Roshan Sedar、Charalampos Kalalas、Francisco Va´zquez-Gallego 和 Jesus Alonso-Zarate。基于强化学习的车载网络错误行为检测。In *ICC 2022 - IEEE International Conference on Communications*, pages 3550-3555, 2022.

[22] Goodness Oluchi Anyanwu、Cosmas Ifeanyi Nwakanma、Jae-Min Lee 和 Dong-Seong Kim。车联网实时位置伪造攻击检测系统。*第 2021 26 届电气和电子工程师学会新兴技术与工厂自动化国际会议（ETFA)*，第 1-4 页，2021 年。

[23] Jing Zhao 和 Ruwu Wang.Fedmix：考虑跨层信息融合和隐私保护的sybil攻击检测系统。*2022 年第 19 届电气和电子工程师学会传感、通信和网络（SECON）国际会议*，第 199-207 页。

[24] Roshan Sedar、Charalampos Kalalas、Jesus Alonso-Zarate、Francisco Va´zquez-Gallego.车联网中的多域拒绝服务攻击：漏洞洞察与检测性能。In *2022 IEEE 8th International Conference on Network Softwarization (NetSoft)*, pages 438-443, 2022.

[25] 何开明、张翔宇、任少清和孙健。图像识别的深度残差学习。In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770-778, 2016.

[26] Leslie N Smith.神经网络超参数的规范方法：*arXiv preprint arXiv:1803.09820*, 2018.

[27] Ilya Loshchilov 和 Frank Hutter.解耦权重衰减正则化》，*arXiv preprint arXiv:1711.05101*, 2017.