

基于 LLM 的多粒度口令分析研究

洪萌, 邱卫东, 王杨德

(上海交通大学网络空间安全学院, 上海 200240)

摘要: 基于口令的认证是常见的身份认证机制。然而, 大规模口令泄露事件时有发生, 表明口令仍面临着被猜测或者盗用等风险。由于口令可以被视作一种特殊的自然语言, 近年来运用自然语言处理技术进行口令分析的研究工作逐渐展开。目前少有工作在大语言模型 (LLM, large language model) 上探究口令文本分词粒度对口令分析效果的影响。为此, 提出了基于 LLM 的多粒度口令分析框架, 总体上沿用预训练范式, 在大量未标记数据集上自主学习口令分布先验知识。该框架由同步网络、主干网络、尾部网络 3 个模块构成。其中, 同步网络模块实现了 char-level、template-level 和 chunk-level 这 3 种粒度的口令分词, 并提取了口令的字符分布、结构、词块组成等特征知识; 主干网络模块构建了通用的口令模型来学习口令组成规律; 尾部网络模块生成了候选口令对目标库进行猜测分析。在 Tianya、Twitter 等 8 个口令库上进行大量实验, 分析总结了多粒度分词下所提框架在不同语言环境中的口令分析效果。实验结果表明, 在中文用户场景中, 基于 char-level 和 chunk-level 分词的框架口令分析性能接近一致, 且显著优于基于 template-level 分词的框架; 在英文用户场景中, 基于 chunk-level 分词的框架口令分析性能最佳。

关键词: 大语言模型; 口令分析; 自然语言处理; 分词

中图分类号: TP309

文献标志码: A

DOI: 10.11959/j.issn.2096-109x.2024008

Research on multi-granularity password analysis based on LLM

HONG Meng, QIU Weidong, WANG Yangde

School of Cyber Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

Abstract: Password-based authentication has been widely used as the primary authentication mechanism. However, occasional large-scale password leaks have highlighted the vulnerability of passwords to risks such as guessing or theft. In recent years, research on password analysis using natural language processing techniques has progressed, treating passwords as a special form of natural language. Nevertheless, limited studies have investigated the impact of password text segmentation granularity on the effectiveness of password analysis with large language models. A multi-granularity password-analyzing framework was proposed based on a large language model, which follows the

收稿日期: 2023-10-19; 修回日期: 2024-01-01

通信作者: 邱卫东, qiuwd@sjtu.edu.cn

基金项目: 国家自然科学基金 (61972249); 国家重点研发计划 (2023YFB3106501)

Foundation Items: The National Natural Science Foundation of China (61972249), The National Key R&D Program of China (2023YFB3106501)

引用格式: 洪萌, 邱卫东, 王杨德. 基于 LLM 的多粒度口令分析研究[J]. 网络与信息安全学报, 2024, 10(1): 112-122.

Citation Format: HONG M, QIU W D, WANG Y D. Research on multi-granularity password analysis based on LLM[J]. Chinese Journal of Network and Information Security, 2024, 10(1): 112-122.

pre-training paradigm and autonomously learns prior knowledge of password distribution from large unlabelled datasets. The framework comprised three modules: the synchronization network, backbone network, and tail network. The synchronization network module implemented char-level, template-level, and chunk-level password segmentation, extracting knowledge on character distribution, structure, word chunk composition, and other password features. The backbone network module constructed a generic password model to learn the rules governing password composition. The tail network module generated candidate passwords for guessing and analyzing target databases. Experimental evaluations were conducted on eight password databases including Tianya and Twitter, analyzing and summarizing the effectiveness of the proposed framework under different language environments and word segmentation granularities. The results indicate that in Chinese user scenarios, the performance of the password-analyzing framework based on char-level and chunk-level segmentation is comparable, and significantly superior to the framework based on template-level segmentation. In English user scenarios, the framework based on chunk-level segmentation demonstrates the best password-analyzing performance.

Keywords: large language model, password analysis, natural language processing, word segmentation

0 引言

口令是保护系统信息的第一道防线,主导着计算机系统的用户身份验证^[1]。随着计算机技术的不断发展,文本口令的许多替代方案涌现出来,如基于指纹和人脸识别的生物识别方案,但这些方案存在技术不成熟、经济代价高等问题。口令凭借可用性和易从丢失中恢复的特性^[2-3],在可预见的未来不会被其他方法取代。

用户注册口令时,根据网站的口令策略设置方便自己记忆的口令。然而,人工生成的口令往往是非随机的弱口令,包含丰富的结构和个人信息,且存在重复使用的情况,极易遭受口令猜测攻击。比如,用户可能设置口令“zhangsan123”,该口令结构简单,且包含用户姓名信息。攻击者观察到人们选择的口令在口令空间中分布不均匀,仅覆盖指数级口令空间的一小部分。多起口令数据库泄露事件表明,用户倾向于选择易于猜测的口令^[4-5],并且主要由常见的字符串(如“password”“12345678”“iloveyou”)及其变体组成。

此外,口令以哈希形式存储于服务器端,以防止在数据库受到破坏时泄露纯文本。在离线猜测攻击的背景下,针对一个账户的用户名,攻击者不断尝试,直到猜测的口令哈希值与服务器端保存的哈希值匹配。通过提升自身掌握的硬件资源、计算资源等,攻击者的猜测能力不断提高。为了积极应对猜测攻击给用户安全带来的威胁与

挑战,针对口令分析的研究持续展开。

就数据驱动模型而言,研究者提出了 OMEN^[6]、FLA^[7]、PCFG^[8]等猜测模型,这些模型都是建立在口令的概率分布^[9]之上的。就基于规则的口令破解工具而言,包含 Hashcat 和 John the Ripper 等在内的工具能够针对目标库有效猜测口令。就口令创建规则而言,除了字符组成和口令结构^[10-13]以外,研究者在口令所包含的语义信息方面进行了探索^[14],捕获了更多的口令组成规律。

近年来,研究者开始采用自然语言处理(NLP, natural language processing)方法来优化口令分析,获得了比传统口令猜测方法更优越的性能。Xu 等^[15]提出了一种基于双向 Transformer 的口令猜测框架 PassBERT,它将“预训练-微调”范式应用于口令猜测攻击。作为 NLP 领域广泛应用的模型,双向 Transformer 模型使得预训练阶段的 PassBERT 充分学习训练集中的口令分布知识。在此基础上,Xu 等^[15]设计了 3 种真实攻击场景下的模型微调方法,分别是条件口令猜测、有针对性的口令猜测和基于规则的自适应口令猜测。实验结果表明,在 PassBERT 框架下,微调后的模型在 3 种攻击中的性能均较为优越,验证了双向 Transformer 对下游口令猜测任务的有效性。

本文利用 NLP 方法进行口令分析,提出了基于大语言模型的结构化口令分析框架。该框架引入 CLT(char-level tokenization)、TLT(template-level tokenization)、CKLT(chunk-level tokenization) 3 种分词方法,研究口令文本分词粒度对口令

令分析效果的影响。同时,利用大语言模型的特性在大量未标记训练集上学习口令分布先验知识。通过在不同用户场景和测试集上开展实验,总结多粒度分词下框架的口令分析性能。实验结果表明,在中文用户场景中,经 CLT、CKLT 方法分词的结构化框架口令分析性能接近一致,且显著优于采用 TLT 方法分词的框架。在英文用户场景中,经 CKLT 方法分词的框架优势凸显。

1 相关工作

1.1 LLM

大语言模型 (LLM, large language model)^[16]是包含数千亿参数的 Transformer 语言模型,旨在理解和生成人类语言。通过在海量文本数据上进行训练^[17-19],LLM 可以执行文本摘要、机器翻译、情感分析等 NLP 任务,表现出强大的自然语言理解和自然语言生成的能力。LLM 对人工智能领域产生了重大影响,ChatGPT 的出现更引发了人们对通用人工智能可能性的重新思考。

本文通过扩展 LLM 中流行的 GPT2^[20]模型实现口令文本学习和生成口令猜测等功能。GPT2 通过访问大量未标记数据集实现无监督学习,获得模型的初始参数。针对输入的一定长度的连续文本序列,GPT2 模型依据第 1 到 i 位的输入预测第 $i+1$ 位的可能值。对语言理解和文本生成等任务,该模型性能较突出。

Rando 等^[21]基于 GPT2 架构提出了一个口令猜测模型 PassGPT,对口令中单个字符的条件分布进行建模。该模型引入了引导口令生成的任务,设置口令长度、模板等限制,控制对每个字符的采样,使得猜测口令的生成过程能满足特定约束。PassGPT 对模型输入只进行单一的字符级分词,但实际上口令具有不同粒度的组合方式^[8,22]。本文在文献[21]的基础上进一步完善,引入多种分词方法,并基于 GPT2 模型探究分词粒度对口令分析效果的影响。

1.2 口令的类语言性

口令的类语言性,使得可以将口令分析视为一项语言建模任务^[23],从而更好地发挥 LLM 的性能。

由具有意义的字符构成的口令可以看作简短

的自然语言,这是因为口令与自然语言存在许多相似之处。一是二者的基本组成单元均为字符;二是二者均具有语义上的连续性。语义上的连续性主要体现在,自然语言中能够根据给定的上文预测下一个隐藏值,在口令中根据给定的口令字符,也能推测出下一位字符的可能值。例如,当给定残缺的句子“My favorite season is [MASK]”时,[MASK]对应的单词只可能为“spring”“summer”“autumn”“winter”。同样地,当给定被遮挡了最后一位的口令“1q2w3e4[MASK]”时,[MASK]对应的字符为“r”的概率远高于“a”或“b”。

1.3 数据集

本文实验所用口令数据集如表 1 所示,其包含约 2.5 亿条口令。其中,Gmail、Twitter、Rockyou、Myheritage 来自英文网站,CSDN、Tianya、178、17173 来自中文网站。这些数据集在互联网公开披露,并被广泛应用于口令的研究工作^[5-8]。

表 1 实验所用口令数据集
Table 1 The password datasets used for experiments

序号	名称	泄露时间	口令数目/个	用途
1	Gmail	2014.09	4 663 677	训练集
2	Twitter	2016.06	67 095 263	测试集
3	Rockyou	2009.12	28 705 927	测试集
4	Myheritage	2018.08	84 825 745	测试集
5	CSDN	2011.12	6 387 785	训练集
6	Tianya	2011.12	30 274 001	测试集
7	178	2011.12	9 072 688	测试集
8	17173	2011.12	17 942 621	测试集

Gmail 提供的是电子邮件服务;Twitter 是一个社交网络服务平台;Rockyou 是一个流行的游戏信息网站;Myheritage 是一个图片服务网站;CSDN 是中文编程论坛;Tianya 是一个大型网络社交平台;178 是一个提供游戏信息的网站;17173 是一个中文游戏门户网站。

1.4 Monte Carlo 模拟

Monte Carlo^[24]模拟以概率论和统计理论方法为基础,通过随机数和概率来解决复杂问题。其核心思想是将实际问题转化为概率模型,通过计算机实现统计模拟,以获得问题的近似解。

OMEN^[6]、FLA^[7]、PCFG^[8]等概率模型,均

采用 Monte Carlo 模拟方法判定给定口令的强度。本文同样借助 Monte Carlo 模拟，在获取大量猜测口令及其概率后，将给定口令的概率值映射为预估猜测次数，从而使得对各测试集的口令覆盖率具备直观的评估指标。

2 方案框架

本文方案分析用户创建口令的方式，对口令进行不同粒度的分割，全面地学习口令的内在特征（字符分布、结构、词块组成等），从多维度理解口令、捕获口令内在组成规律后，可以把学到的知识用于指导口令的生成和猜测攻击的升级，从而保护用户的个人隐私和信息安全。

对口令文本进行建模和分析，本质是寻求对口令分布的最大似然估计。LLM 能从大量文本中对自然语言进行建模，天然地适合口令猜测任务。基于 LLM 的结构化口令分析框架，总体上沿用的是 NLP 中的预训练范式，从大量的未标记训练集中提取口令文本的共性特征。基于 LLM 的结构化口令分析框架如图 1 所示，分为同步网络、主干网络和尾部网络 3 个模块。

2.1 同步网络

用户设置口令有自己的技巧，统计表明口令中经常出现如“ing”“123”“520”等词根。如果一味以单个字符为单位分割口令，每次都把“ing”词根分词为“i”“n”“g”输入模型训练，必然导致计算资源的巨大浪费。因而，口令的精细组合是值得研究的课题。

同步网络完成对输入的预处理，将其转化成能被识别的标记（token）。该模块包含 CLT、TLT、CKLT 等多种分词方法，实现从口令字符串到不同粒度口令组合的转换，并提取了多种口令内在特征，不同分词方法如表 2 所示。本文仅关注英

语，不考虑法语、西班牙语等其他语种。

表 2 不同分词方法
Table 2 Different segmentation methods

分词方法	分词级别	分词粒度	特征提取
CLT	char level	较细	字符分布
TLT	template level	较粗	结构
CKLT	chunk level	中等	词块组成

2.1.1 CLT 方法

基于 char-level 的分词方法，把口令视作单个字符序列，以单个字符为单位分割口令，分词粒度较细。该方法旨在学习口令的字符分布特征（口令长度、字符种类）。

任意输入被分割为序列 $\{x_1, x_2, \dots, x_{n-1}, x_n\}$ ，每个字符 x_i 的概率由它前面的字符决定，局部条件概率为 $P(x_i | x_1, \dots, x_{i-1})$ 。此时口令的概率计算如下。

$$P(\mathbf{x}) = P(x_1 x_2 \dots x_{n-1} x_n) = \prod_{i=1}^n P(x_i | x_1, \dots, x_{i-1}) \quad (1)$$

其中，字符包括 95 个可打印 ASCII 字符和 5 个特殊字符[UNK]、[SEP]、[CLS]、[PAD]、[MASK]，以此构成了长度为 100 的 CLT 词表。[UNK]标识未识别的字符，[SEP]标识口令的末尾位，[CLS]标识口令的初始位，[PAD]用于补齐短口令（本文默认分词序列最大长度为 16），[MASK]用于遮盖口令中的一部分。

以“happy888”为例，该口令经 CLT 方法分词为 $\{h, a, p, p, y, 8, 8, 8\}$ 。

2.1.2 TLT 方法

基于 template-level 的分词方法，本质是从口令的结构出发理解口令，分词粒度较粗。参照概率上下文无关文法^[8]，该方法能提取口令的结构特征。

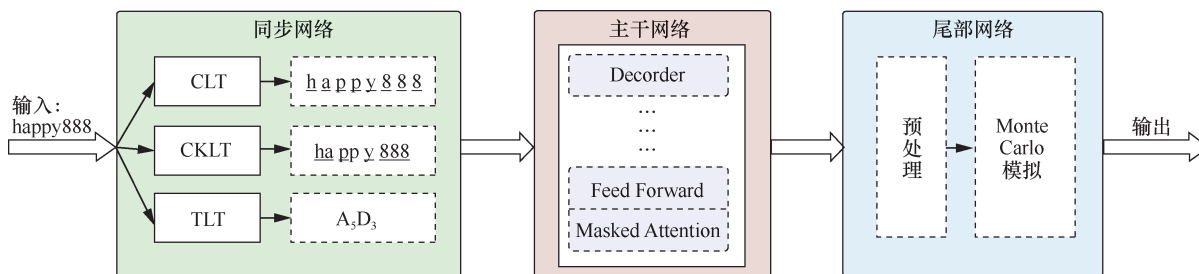


图 1 基于 LLM 的结构化口令分析框架
Figure 1 LLM-based structured password analysis framework

已知口令是由字符构成的序列，字符通常包括字母、数字和特殊符号 3 种类别，分别标记为 A (alpha)、D (digit)、S (symbol)。从组成结构出发，口令可以看作由一定长度的连续字母序列、连续数字序列和连续特殊符号序列拼接而成。结合序列的长度信息，可推导出每条口令的结构，形如 “ $A_X D_X S_X$ ”。比如口令 “happy888”，经过 TLT 方法，它被分词为 $\{A_5 D_3\}$ 。

运用 TLT 方法，除了能从训练样本中获得大量口令结构信息以外，还能统计计数得到不同长度字母、数字和特殊符号序列的条件概率，这些概率将在尾部网络生成猜测口令时发挥作用。表 3 至表 5 展示了一次 TLT 分词得到的部分连续字母序列、连续数字序列、连续特殊符号序列及其概率。

表 3 部分连续字母序列
Table 3 Partial continuous alpha sequence

字符串	概率	字符串	概率
qq	0.060 1	wang	0.031 3
aa	0.047 8	love	0.025 7
li	0.018 8	chen	0.023 3
as	0.013 8	yang	0.020 0
ab	0.012 1	abcd	0.017 8

表 4 部分连续数字序列
Table 4 Partial continuous digit sequence

字符串	概率	字符串	概率
12	0.069 7	1234	0.082 1
11	0.061 5	1314	0.064 2
88	0.054 1	1111	0.012 1
00	0.052 0	0000	0.007 5
99	0.033 0	8888	0.006 4

表 5 部分连续特殊符号序列
Table 5 Partial continuous symbol sequence

字符串	概率	字符串	概率
..	0.261 9	!@#\$	0.137 0
**	0.084 6		0.116 7
@@	0.052 0	+-* /	0.062 7
++	0.047 4	****	0.048 1
!!	0.039 8	/*-+	0.033 5

由 4 个字母和 4 位数字组成的结构表示为 “ $A_4 D_4$ ”，满足此结构的口令 “love1998” 概率计

算如下。

$$P(\text{love1998}) = P(A_4 D_4) P(\text{love} | A_4) P(1998 | D_4) \quad (2)$$

一般来说，在 TLT 方法中，每条口令的概率为口令结构的概率与该结构中字母、数字和特殊符号序列条件概率的乘积，表示为

$$P(\mathbf{x}) = P(x_1 x_2 \cdots x_{n-1} x_n) = P(\text{seq}_1 \text{seq}_2 \cdots \text{seq}_r) \cdot \prod_{i=1}^r P(\text{pwd}_i | \text{seq}_i) \quad (3)$$

2.1.3 CKLT 方法

BPE^[25]算法是一种简单的数据压缩技术，主要思想是利用子词来编码数据，在降低词表大小的同时能学到单词语义信息，该算法的时间复杂度较低。通过改造 BPE 算法，CKLT 方法对训练集实现了基于 chunk-level 的分词^[22]，学习口令的词块组成特征，分词粒度介于 CLT 方法和 TLT 方法之间。

词块是经常一起出现的相关字符序列，它从数据集内部统计获得，多个词块共同组成了口令。例如，口令 “iloveu4ever” 可以分为词块 “iloveu” “4ever”。

任意输入被分割为序列 $\{\text{chk}_1, \text{chk}_2, \cdots, \text{chk}_m\}$ ，其中每个词块 chk_i 的概率由它前面的词块决定，局部条件概率为 $P(\text{chk}_i | \text{chk}_1, \cdots, \text{chk}_{i-1})$ 。此时口令的概率计算为

$$P(\mathbf{x}) = P(x_1 x_2 \cdots x_{n-1} x_n) = P(\text{chk}_1 \text{chk}_2 \cdots \text{chk}_m) = \prod_{i=1}^m P(\text{chk}_i | \text{chk}_1, \cdots, \text{chk}_{i-1}) \quad (4)$$

图 2 是 CKLT 分词过程示例。CKLT 方法的原理在于：在训练集内学习获得词块词汇表，基于词块词汇表将口令分段。其中，获取词块词汇表的主要步骤如下。

Step 1 准备口令训练集。

Step 2 设置词汇表大小 (vocab_size) 超参数，作为算法终止的条件。

Step 3 在每个口令末尾添加停止符 <w>，并统计每个口令出现的概率，如图 2 中口令 “password88” 出现 4 次，将其标记为 “password88<w>: 4”。

Step 4 将所有口令拆分为单个字符，以所

有单个字符建立最初的词汇表，并统计每个字符的出现频次，此时，口令“password88”可以标记为“password88</w>:4”。

Step 5 挑出频次最高的口令字符对合并，将合并后的字符对加入词汇表，并将训练集中所有该字符对融合，在图 2 中，首先将出现频次最高的字符对“wo”合并为“wo”。

Step 6 重复 Step5，直到词汇表大小达到 Step2 设置的初始值。

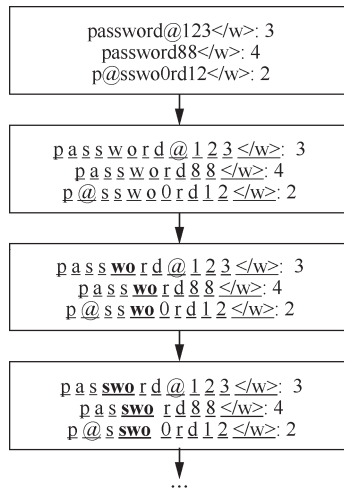


图 2 CKLT 分词过程示例

Figure 2 Example of CKLT word segmentation process

一次 CKLT 分词尝试得到的部分词块如表 6 所示，对表中词块的长度进行统计分析，此时词块平均长度 (avg_len) 为 2.93。

表 6 部分词块结果
Table 6 Results of partial chunk

序号	词根	序号	词根	序号	词根
1	y	7	..	13	xy
2	888	8	jian	14	aini
3	27	9	111	15	gh
4	1990	10	19830	16	02
5	am	11	152	17	ha
6	pp	12	ke	18	1234567890

同样以口令“happy888”为例，经 CKLT 方法分词结果为{ha pp y 888}。

2.2 主干网络

主干网络用于捕捉口令字符的组成规律，建模口令字符转移概率。本文扩展 LLM 来构建主

干网络模块。

口令中字符的分布实际上差异很大，口令分析是一项不平衡的分类任务。LLM 强大的自然语言建模能力，能学习通用的表示形式，具有建模口令分布的潜力。然而，口令文本的单向性限制了模型架构的选择。Bert^[26]、XLNet^[27]等 NLP 模型学习上下文表示，可能带来可解释性差等问题。GPT2^[20]模型使用受限自注意力，即每个标记只能关注其左侧的内容，右侧是未知的，可以很好地适应口令猜测任务。

主干网络结构如图 3 所示。

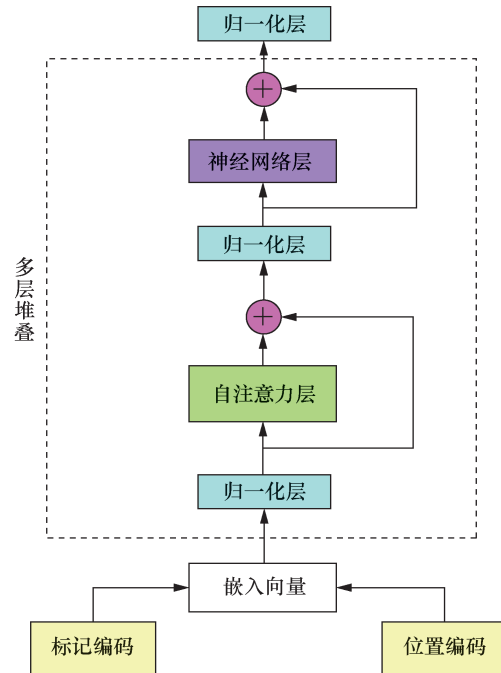


图 3 主干网络结构

Figure 3 Backbone network structure

从图 3 中可以看出，主干网络的本质是多层 Transformer 解码器结构，将口令对应的向量表示输入模型后，首先通过自注意力层处理，接着将其传递给神经网络层。第一层模块处理完成后，将结果向量传入下一层 Transformer 模块继续进行计算。每一层 Transformer 模块的处理方式相同，但每一层都会维护自己的自注意力层和神经网络层中的权重。

考虑到口令具有文本短、上下文关系简单的特性，为了平衡计算开销和模型性能，本文设置多组模型参数，对比不同大小模型的口令分析效果。以 Gmail 为训练集，猜测分析 Twitter 口令库。当猜测 10^4 次时，模型参数对照如表 7 所示。

表 7 模型参数对照
Table 7 Model parameters comparison

模型规模	解码器模块/层	自注意力头的数目	隐藏层大小	训练参数	覆盖率
Small	6	256	4	4 781 056	79.98%
Base	12	256	12	85 934 712	80.92%

实验结果表明, 小规模模型足够充分学习口令的内在特征, 更大规模的语言模型虽然能提升口令分析效果, 但收效甚微。因此, 本文设置 Transformer 解码器模块为 6 层, 自注意力头的数目为 256, 隐藏层大小为 4, 此时模型大小约为 18 MB。

2.3 尾部网络

尾部网络分为预处理和 Monte Carlo 模拟两个部件。预处理部件负责从主干网络采样生成口令猜测。Monte Carlo 模拟部件结合猜测口令, 解析得到目标库中每条口令的预估猜测次数。

特别的是, 采用 TLT 方法分词时, 输入主干网络模型的是形如 “ $A_X D_X S_X$ ” 的口令结构, 因此从主干网络采样生成的是口令结构及其概率。部分口令结构采样结果如表 8 所示。为进一步生成口令猜测、解析口令概率, 预处理部件需结合式(3)的计算原理和 2.1.2 节中提到的字符序列概率分布进行计算。

表 8 部分口令结构采样结果
Table 8 Results of partial password structure sampling

序号	口令结构	概率
1	D_8	0.220 8
2	D_9	0.114 9
3	A_8	0.044 9
4	D_{11}	0.041 8
5	D_{10}	0.037 4

3 实验结果与分析

3.1 数据集准备

由于大部分网站口令策略对口令长度的要求为 6 位至 20 位, 为保证训练集数据的质量, 在实验开始前对 8 个数据集进行清洗。具体做法是: 删除含有非可打印 ASCII 码字符和长度小于 6 位或者超过 16 位的异常口令, 并在各数据集中随机抽取 100 万条口令作为数据集样本。

多粒度分词下基于 LLM 的口令分析实验,

以 CSDN、Gmail 分别作为中文用户场景和英文用户场景的训练集, 得到 CN-CLT、CN-CKLT、CN-TLT、EN-CLT、EN-CKLT、EN-TLT 这 6 个预训练框架。为评估上述框架的口令分析性能, 本文以 178、17173、Tianya 作为中文测试集, 以 Twitter、Rockyou、Myheritage 作为英文测试集。

3.2 实验设计

本文实验的设计围绕以下 4 个问题展开。

问题 1 相较于其他口令猜测模型, GPT2 模型的性能如何?

问题 2 相较于传统基于概率上下文无关文法的模板式口令分析方法, 基于 LLM 的 TLT 方法性能如何?

问题 3 在 CKLT 方法中, 超参数 vocab_size 是否会对口令分析效果产生影响?

问题 4 针对不同语种用户, 多粒度分词下框架的口令分析效果如何?

3.3 实验结果

3.3.1 CN-CLT、EN-CLT 框架与 FLA 模型的性能对比

为评估 GPT2 模型的性能, 本小节对比了基于 char-level 分词的框架 CN-CLT、EN-CLT 与其他口令猜测模型。

典型的字符级口令猜测模型包括 OMEN^[6]、FLA^[7]、PassGAN^[28]等, 但是 PassGAN 等模型无法采用 Monte Carlo 模拟方法分析各测试集, 不符合对比实验的需求。因此, 本节选取性能较突出的 FLA 模型用于对比实验。该模型基于循环神经网络, 具有可压缩性和高效性。

以 CSDN、Gmail 作为 FLA 模型在中英文用户场景中的训练集, 随后猜测分析各中英文目标库 (以 178、Rockyou 库为例)。CN-CLT、EN-CLT 框架与 FLA 模型的 Monte Carlo 模拟结果 (即覆盖率) 如图 4 所示。

从图 4 可以看出, 当猜测次数小于 10^{10} 次时, FLA 模型对测试集的覆盖率更高。随着猜测次数

的增加, CN-CLT、EN-CLT 框架对测试集的覆盖率逐渐高于 FLA 模型。该结果表明, 在大规模猜测场景中, GPT2 模型的性能优于 FLA 模型。

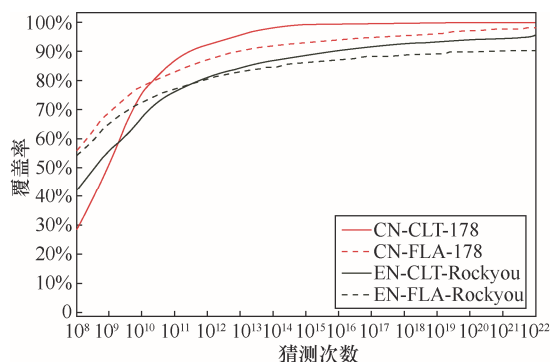


图4 CN-CLT、EN-CLT 框架与 FLA 模型的 Monte Carlo 模拟结果
Figure 4 Monte Carlo simulation results of CN-CLT, EN-CLT framework and FLA model

3.3.2 CN-TLT、EN-TLT 框架与 PCFG 模型的性能对比

为探究在 LLM 中引入 TLT 方法的性能, 本节对比了本文所提框架与 PCFG^[8]模型。由 2.1.2 节可知, CN-TLT、EN-TLT 框架与 PCFG 模型都采用基于 template-level 的分词方法, 将口令映射为它所对应的结构。不同的是, PCFG 模型从训练集中推导出口令结构及其概率分布, 而本文方法是基于 LLM 学习推导得到的口令结构, 随后采样生成口令结构的猜测和概率分布。

同样以 CSDN、Gmail 作为 PCFG 模型在中英文用户场景中的训练集, 随后猜测分析各中英文目标库, 将得到的结果与 CN-TLT、EN-TLT 框架的 Monte Carlo 模拟结果进行比较, 如图 5、图 6 所示。

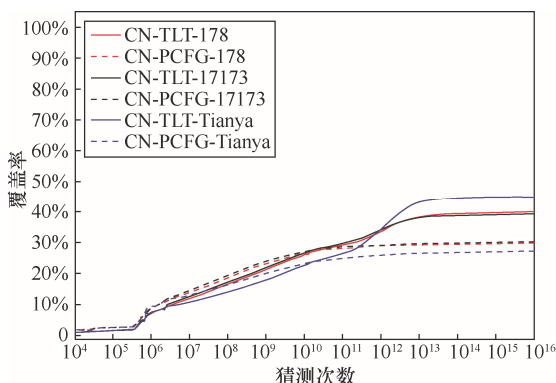


图5 CN-TLT 框架与 PCFG 模型的 Monte Carlo 模拟结果
Figure 5 Monte Carlo simulation results of CN-TLT framework and PCFG model

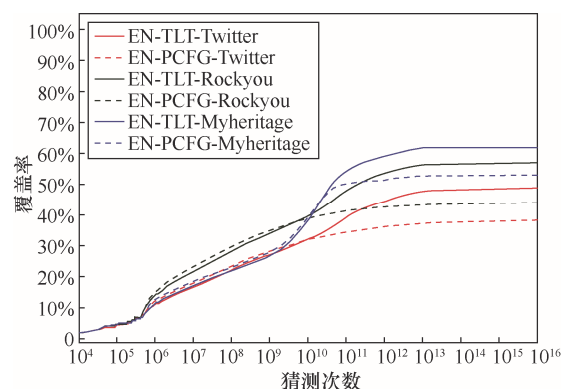


图6 EN-TLT 框架与 PCFG 模型的 Monte Carlo 模拟结果
Figure 6 Monte Carlo simulation results of EN-TLT framework and PCFG model

由图 5、图 6 可以得出, 总体来说, 经 TLT 方法分词的 CN-TLT、EN-TLT 框架性能优于 PCFG 模型。猜测次数较少时, PCFG 模型的性能稍稍优于 CN-TLT、EN-TLT 框架。随着猜测次数的增加, CN-TLT、EN-TLT 框架的优势逐渐显现。当猜测 10^{16} 次时, 对各中文测试集的覆盖率, CN-TLT 框架平均高于 PCFG 模型 13.1%; 对各英文测试集的覆盖率, EN-TLT 框架平均高于 PCFG 模型 12.3%。

虽然 CN-TLT、EN-TLT 框架与 PCFG 模型分词原理相同, 但是基于 LLM 学习口令结构可以实现更高效的口令猜测。这是因为 LLM 强大的自然语言理解和自然语言生成能力, 能充分地学习口令结构特征, 采样生成训练集中尚未出现的结构样式。清洗不符合“ $A_x D_x S_x$ ”范式的异常结构后, 由新生成的口令结构能生成更多口令猜测, 从而达到更高的猜测效率。

3.3.3 关于不同精度 CKLT 方法的实验

由 2.1.3 节分词原理可知, vocab_size 是影响 CKLT 词根合并次数的重要参数, 决定了 CKLT 方法的分词精度, avg_len 可以用来区分不同精度的 CKLT 分词结果。

为分析超参数 vocab_size 对口令分析效果的影响, 在 CSDN 数据集上进行多次 chunk-level 分词。尝试将 vocab_size 设置为 130、200、500、20 000 共 4 组值, 此时 avg_len 值依次为 1.5、1.8、2.5、4.5, 预训练得到的口令分析框架标记为 CN-CKLT-1.5、CN-CKLT-1.8、CN-CKLT-2.5、CN-CKLT-4.5。以 Tianya 口令集作为目标库, 得到的 Monte Carlo 模拟结果如图 7 所示。

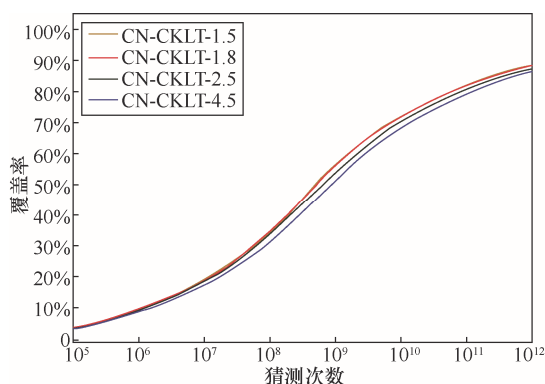


图 7 不同框架分析 Tianya 口令集的 Monte Carlo 模拟结果
Figure 7 Monte Carlo simulation results of different frameworks analyzing Tianya dataset

从图 7 可以看出, CN-CKLT-1.5 框架和 CN-CKLT-1.8 框架对 Tianya 口令集的覆盖率几乎相等, 且高于其他框架。因此, 在兼顾词根多样性的基础上, 本文选取 1.8 作为 `avg_len` 的最佳值。即在后续实验中, 认为当词块平均长度为 1.8 时, 基于 `chunk-level` 分词的框架能取得最佳的口令分析效果。

3.3.4 关于不同语种用户的实验

图 8~图 10 依次为 CN-CLT、CN-CKLT、CN-TLT 框架猜测 178、17173、Tianya 口令集的 Monte Carlo 模拟结果。

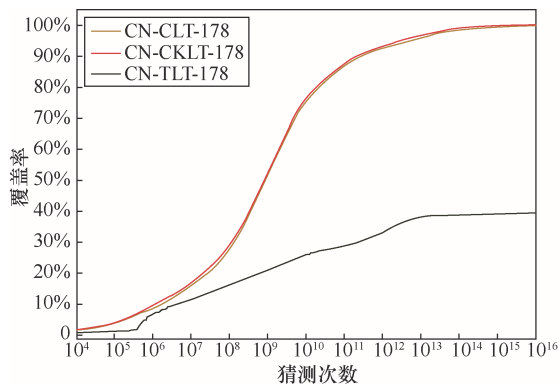


图 8 中文用户场景中分析 178 口令集的 Monte Carlo 模拟结果
Figure 8 Monte Carlo simulation results of analyzing 178 password dataset in the Chinese user scenario

从图 8~图 10 可以得出, 在中文用户场景中, CN-CLT、CN-CKLT 框架性能接近一致, 且显著优于 CN-TLT 框架。当猜测次数较少时, CN-CLT、CN-CKLT 框架的性能稍稍优于 CN-TLT 框架。随着猜测次数的增加, CN-CLT、CN-CKLT 框架的领先优势持续扩大。当口令猜测次数达到 10^{13} 时,

对测试集的覆盖率, CN-CLT 框架平均高于 CN-TLT 框架 53.7%。

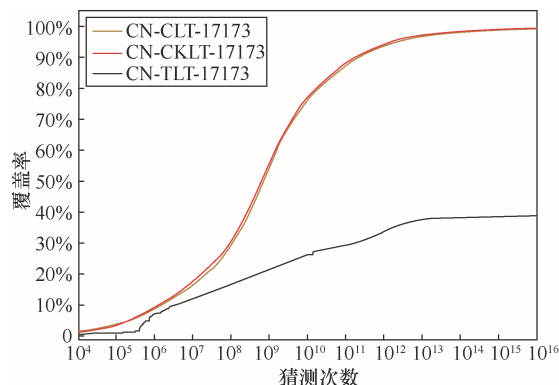


图 9 中文用户场景中分析 17173 口令集 Carlo 模拟结果
Figure 9 Monte Carlo simulation results of analyzing 17173 password dataset in the Chinese user scenario

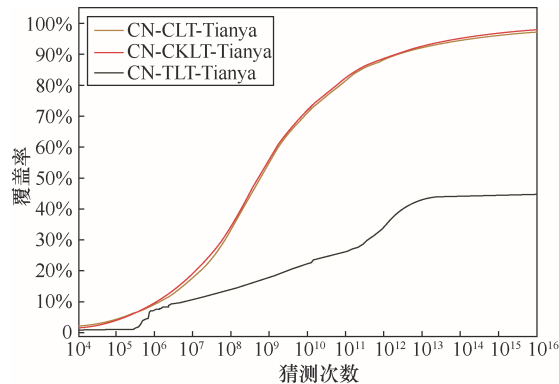


图 10 中文用户场景中分析 Tianya 口令集的 Monte Carlo 模拟结果
Figure 10 Monte Carlo simulation results of analyzing Tianya password dataset in the Chinese user scenario

图 11~图 13 依次为 EN-CLT、EN-CKLT、EN-TLT 框架猜测 Twitter、Rockyou、Myheritage 口令集的 Monte Carlo 模拟结果。

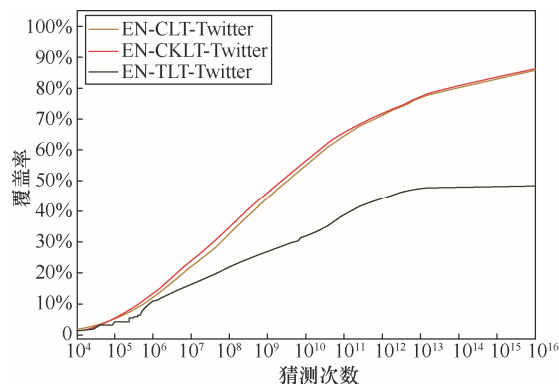


图 11 英文用户场景中分析 Twitter 口令集的 Monte Carlo 模拟结果
Figure 11 Monte Carlo simulation results of analyzing Twitter password dataset in the English user scenario

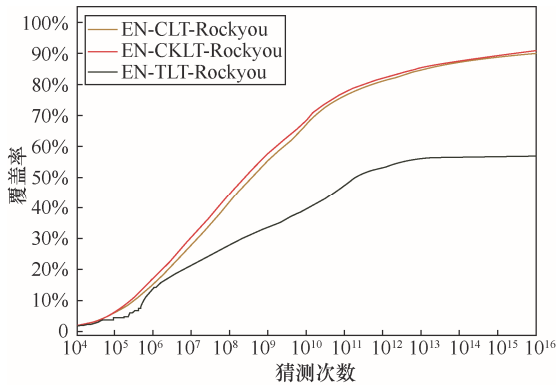


图 12 英文用户场景中分析 Rockyou 口令集的 Monte Carlo 模拟结果
Figure 12 Monte Carlo simulation results of analyzing Rockyou password dataset in the English user scenario

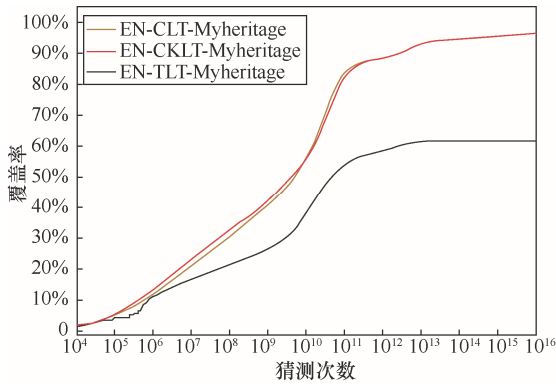


图 13 英文用户场景中分析 Myheritage 口令集的 Monte Carlo 模拟结果
Figure 13 Monte Carlo simulation results of analyzing Myheritage password dataset in the English user scenario

从图 11~图 13 可以得出，英文用户场景中，EN-CKLT 框架的口令分析性能最佳，代表 EN-CKLT 框架的曲线几乎始终处于最上方。当猜测次数为 10^8 时，对测试集的覆盖率，EN-CKLT 框架平均高于 EN-CLT 框架 3.1%，平均高于 EN-TLT 框架 14.3%。

结合图 8~图 13 发现，与中文用户场景相比，在英文用户场景中，使用 CKLT 方法分词会使口令猜测效率显著提升。这可能是因为相较于 CSDN 训练集，从 Gmail 训练集中提取的词块质量更高，对分割口令更有效。

4 结束语

本文将自然语言处理技术应用于口令分析，提出了基于 LLM 的结构化口令分析框架。结构化口令分析框架由同步网络、主干网络、尾部网络 3 个模块构成。在该框架基础上，实现了 char-level、template-level、chunk-level 这 3 种粒

度的口令分词方法，并在构建的预训练模型中学习口令字符转移概率和组成规律。在多语种口令库和大规模猜测的场景中，生成大量候选口令对测试集进行猜测分析，最后总结出不同粒度分词方法对口令分析性能的影响。实验结果表明，在中文用户场景中，基于 char-level 和 chunk-level 分词的框架口令分析性能接近一致，且显著优于基于 template-level 分词的框架。在英文用户场景中，基于 chunk-level 分词的框架具有绝对优势。

结构化的思想使得同步网络、主干网络和尾部网络 3 个模块相对独立，功能相对专一，可以满足不同口令攻击场景的需求，口令分析的复用性得到提升。此外，随着研究的深入，新涌现的分词方法可以便利地扩展到框架中，以开展更多攻击场景和猜测条件下的拓展实验。

本文方案仍然存在一定的局限性。一方面，实验所用训练集为包含 100 万条口令的子集，更大数据集可能会对实验结果产生影响。另一方面，实验过程中仅设计了中文、英文两种场景评估口令分析框架的性能，评估标准过于单一。

参考文献：

- [1] BONNEAU J, HERLEY C, OORSCHOT P C V, et al. The quest to replace passwords: a framework for comparative evaluation of web authentication schemes[C]//Proceedings of 2012 IEEE Symposium on Security and Privacy. 2012: 553-567.
- [2] BONNEAU J, HERLEY C, VAN OORSCHOT P C, et al. Passwords and the evolution of imperfect authentication[J]. Communications of the ACM, 2015, 58(7): 78-87.
- [3] YAN J, BLACKWELL A, ANDERSON R, et al. Password memorability and security: empirical results[J]. IEEE Security & Privacy, 2004, 2(5): 25-31.
- [4] DELL'AMICO M, MICHIARDI P, ROUDIER Y. Measuring password strength: an empirical analysis[J]. arXiv preprint arXiv: 0907.3402, 2009.
- [5] MA J, YANG W N, LUO M, et al. A study of probabilistic password models[C]//Proceedings of 2014 IEEE Symposium on Security and Privacy. 2014: 689-704.
- [6] DÜRMUTH M, ANGELSTORF F, CASTELLUCCIA C, et al. OMEN: faster password guessing using an ordered Markov enumerator[C]//Proceedings of International Symposium on Engineering Secure Software and Systems. 2015: 119-132.
- [7] MELICHER W, UR B, SEGRETINI S M, et al. Fast, lean, and accurate: modeling password guess ability using neural networks[C]//Proceedings of the 25th USENIX Conference on Security Symposium. 2016: 175-191.
- [8] WEIR M, AGGARWAL S, MEDEIROS B D, et al. Password

- cracking using probabilistic context-free grammars[C]//Proceedings of 2009 30th IEEE Symposium on Security and Privacy. 2009: 391-405.
- [9] PASQUINI D, GANGWAL A, ATENIESE G, et al. Improving password guessing via representation learning[C]//Proceedings of 2021 IEEE Symposium on Security and Privacy (SP). 2021: 1382-1399.
- [10] NARAYANAN A, SHMATIKOV V. Fast dictionary attacks on passwords using time-space tradeoff[C]//Proceedings of the 12th ACM Conference on Computer and Communications Security. 2005: 364-372.
- [11] CASTELLUCCIA C, DÜRMUTH M, PERITO D. Adaptive password-strength meters from Markov models[C]//Proceedings of NDSS. 2012.
- [12] CHOU H C, LEE H C, YU H J, et al. Password cracking based on learned patterns from disclosed passwords[J]. IJICIC, 2013, 9(2): 821-839.
- [13] SHAY R, KOMANDURI S, KELLEY P G, et al. Encountering stronger password requirements: user attitudes and behaviors[C]//Proceedings of the Sixth Symposium on Usable Privacy and Security. 2010: 1-20.
- [14] VERAS R, COLLINS C, THORPE J. On semantic patterns of passwords and their security impact[C]//Proceedings of NDSS. 2014.
- [15] XU M, YU J, ZHANG X, et al. Improving real-world password guessing attacks via bi-directional transformers[C]//Proceedings of 32nd USENIX Security Symposium (USENIX Security 23). 2023: 1001-1018.
- [16] ZHAO W X, ZHOU K, LI J, et al. A survey of large language models[J]. arXiv preprint arXiv:2303.18223, 2023.
- [17] SHANAHAN M. Talking about large language models[J]. arXiv preprint arXiv:2212.03551, 2022.
- [18] QIU X P, SUN T X, XU Y G, et al. Pre-trained models for natural language processing: a survey[J]. Science China Technological Sciences, 2020, 63(10): 1872-1897.
- [19] HAN X, ZHANG Z Y, DING N, et al. Pre-trained models: past, present and future[J]. AI Open, 2021, 2: 225-250.
- [20] RADFORD A, WU J, CHILD R, et al. Language models are unsupervised multitask learners[J]. OpenAI Blog, 2019, 1(8): 9.
- [21] RANDO J, PÉREZ-CRUZ F, HITAJ B. PassGPT: password modeling and (guided) generation with large language models[J]. arXiv preprint arXiv: 2306.01545, 2023.
- [22] XU M, WANG C W, YU J T, et al. Chunk-level password guessing: towards modeling refined password composition representations[C]//Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. 2021: 5-20.
- [23] LI H, CHEN M Q, YAN S B, et al. Password guessing via neural language modeling[M]//Machine Learning for Cyber Security. 2019: 78-93.
- [24] DELL'AMICO M, FILIPPONE M. Monte Carlo strength evaluation: fast and reliable password checking[C]//Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security. 2015: 158-169.
- [25] GAGE P. A new algorithm for data compression[J]. C Users Journal, 1994, 12(2): 23-38.
- [26] DEVLIN J, CHANG M W, LEE K, et al. Bert: pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.
- [27] YANG Z, DAI Z, YANG Y, et al. XLNet: generalized autoregressive pretraining for language understanding[J]. Advances in Neural Information Processing Systems, 2019, 32.
- [28] HITAJ B, GASTI P, ATENIESE G, et al. PassGAN: A deep learning approach for password guessing[M]//Applied Cryptography and Network Security. Cham: Springer International Publishing, 2019: 217-237.

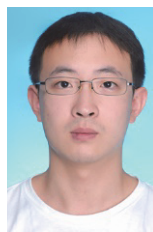
[作者简介]



洪萌（1995—），女，江苏南通人，上海交通大学硕士生，主要研究方向为密码分析。



邱卫东（1973—），男，江西修水人，上海交通大学教授、博士生导师，主要研究方向为计算机取证、密码分析、人工智能安全、大数据隐私保护。



王杨德（1987—），男，辽宁沈阳人，上海交通大学博士生，主要研究方向为计算机取证、密码分析。