基于深度学习的 ABAC 访问控制策略自动化生成技术

刘敖迪[1,2]，杜学绘[1,2]，王娜[1,2]，乔蕊[1,3]

（1. 信息工程大学，河南 郑州 450001；2. 河南省信息安全重点实验室，河南 郑州 450001；
3. 周口师范学院，河南 周口 466001）

摘 要：针对访问控制策略的自动化生成问题，提出了一种基于深度学习的 ABAC 访问控制策略生成框架，从自然语言文本中提取基于属性的访问控制策略，该技术能够显著降低访问控制策略生成的时间成本，为访问控制的实施提供有效支持。将策略生成问题分解为访问控制语句识别和访问控制属性挖掘两项核心任务，分别设计了 BiGRU-CNN-Attention 和 AM-BiLSTM-CRF 这 2 个神经网络模型来实现访问控制策略语句识别和访问控制属性挖掘，从而生成可读、可执行的访问控制策略。实验结果表明，与基准方法相比，所提方法具有更好的性能。特别是在访问控制策略语句识别任务中平均 F1-score 指标能够达到 0.941，比当前的 state-of-the-art 方法性能提高了 4.1%。

关键词：访问控制；ABAC 模型；策略生成；自然语言处理；深度学习

中图分类号：TP309

文献标识码：A

doi: 10.11959/j.issn.1000−436x.2020212

ABAC access control policy generation technique based on deep learning

LIU Aodi[1,2], DU Xuehui[1,2], WANG Na[1,2], QIAO Rui[1,3]

1. Information Engineering University, Zhengzhou 450001, China
2. He’nan Province Key Laboratory of Information Security, Zhengzhou 450001, China
3. Zhoukou Normal University, Zhoukou 466001, China

Abstract: To solve the problem of automatic generation of access control policies, an access control policy generation framework based on deep learning was proposed. Access control policy based on attributes could be generated from natural language texts. This technology could significantly reduce the time cost of access control policy generation and provide effective support for the implementation of access control. The policy generation problem was decomposed into two core tasks, identification of access control policy sentence and access control attribute mining. Neural network models such as BiGRU-CNN-Attention and AM-BiLSTM-CRF were designed respectively to realize identification of access control policy sentence and access control attribute mining, so as to generate readable and executable access control policies. Experimental results show that the proposed method has better performance than the benchmark method. In particular, the average F1-score index can reach 0.941 in the identification task of access control policy sentence, which is 4.1% better than the current state-of-the-art method.

Key words: access control, ABAC model, policy generation, natural language processing, deep learning

---

收稿日期：2020–07–23；**修回日期**：2020–09–30

基金项目：国家重点研发计划基金资助项目（No.2018YFB0803603，No.2016YFB0501901）；国家自然科学基金资助项目（No.61802436, No.61902447）

Foundation Items: The National Key Research and Development Program of China (No.2018YFB0803603, No.2016YFB0501901), The National Natural Science Foundation of China (No.61802436, No.61902447)

1　引言

不断发展的大数据、云计算等新型计算范式极大地提高了数据共享与利用的效率，通过分析与利用数据资源，能够创造出巨大的社会价值和经济价值。然而，数据的共享与利用也面临着严峻的安全风险，导致各类安全事故频发。例如，2018 年 3 月曝出的 Facebook 数据外泄事件，导致超过 5 000 万用户的个人数据被非法访问。因此，数据的非授权共享将会对用户数据带来巨大的安全威胁，实现安全、可控的数据资源共享与利用是数据应用及发展的前提与基础。作为保护数据安全的重要手段之一，访问控制技术[1]能够通过对用户权限的管理，使合法用户依照其所拥有的权限访问系统内的相应资源，禁止非法用户对资源的非授权访问，从而有效地保障数据安全及业务系统的正常运转。其中，基于属性的访问控制机制（ABAC, attribute based access control）[2-3]使用属性作为访问控制的基本要素，能够灵活利用实体所拥有的属性集合来决定是否赋予其访问权限，具有较强的语义表达能力，且兼容自主访问控制、强制访问控制、基于角色的访问控制等机制[4]，适用于解决开放计算环境中的细粒度访问控制和大规模动态授权问题。Gartner 预测[5]到 2020 年，70%的企业将使用基于属性的访问控制方案作为主导机制来保护内部关键信息资产。

访问控制策略是执行访问控制机制的核心与基础[6-7]。特别是在信息系统建立初期，如何在满足系统安全需求的前提下，配置正确、完备且一致的访问控制策略是安全管理人员对资源实施访问控制的前提[3]。现有的策略生成技术主要包括自上向下与自下向上 2 种模式[8-10]。其中，自上向下模式[11]依靠安全专家的专业知识，从系统的业务需求和安全需求出发，通过人工分析的方式来得到系统访问控制策略。但该模式是一项需要专业知识且容易出错的劳动密集型工作，策略生成质量不稳定，可靠性和准确性与安全专家的专业水平直接相关[1-2]。并且，针对不同业务系统，自上向下模式难以移植，可扩展能力较弱，容易导致过度授权和授权不足现象的发生[2]。与自上向下模式不同，自下向上模式[12-15]依据信息系统中已有的访问控制策略信息（用户–权限关系），利用数据挖掘等手段实现策略的自动生成，减少了对专家的依赖。但是，该模式需要依据信息系统中已存在访问控制策略信息（用户–权限关系）作为前置条件，才能实现访问控制策略的生成[16-17]。而在信息系统建立的初期阶段，由于系统中没有现成的访问控制信息作为基础，导致该模式在此场景下难以直接应用。并且，现有自下向上技术大多通过角色[18]来构建策略，得到的角色信息通常是无语义信息，难以与真实世界中访问控制需求相结合，无法表达出 ABAC 模型丰富的属性语义信息（主体属性、动作属性、客体属性），难以直接应用到 ABAC 的策略生成工作中。

实际上，在大多数组织机构的信息系统内部，都存在着以自然语言形式描述的系统项目规范类文档（如项目需求文档、用户手册、使用须知等）。这些项目文档[19]中蕴含了系统预置的与访问控制相关的策略信息，它们是安全专家了解应用环境与应用背景、分析安全需求的重要依据和信息来源。手动筛选现有文档以提取隐藏的访问控制策略可能是一项冗长、耗时且容易出错的工作，且需要具有专业安全知识的专家才能够顺利完成。因此，如何从项目规范类文档中提取访问控制策略信息，自动化生成 ABAC 策略，对基于属性的访问控制研究具有重要意义[20]。

为了解决上述访问控制策略生成的难题，本文提出了一种新颖的基于深度学习的 ABAC 访问控制策略生成技术，目的是从自然语言形式描绘的项目规范类文档中提取出基于属性的访问控制策略，实现系统访问控制策略的自动化、智能化生成，显著降低访问控制策略生成的时间成本，从而为访问控制的实施提供支撑。本文将访问控制策略生成问题分解为访问控制策略语句识别和访问控制属性挖掘 2 项关键任务。其中，访问控制策略语句识别任务是从项目相关文档中提取出与访问控制相关的语句，访问控制属性挖掘任务则是从自然语言形式的策略语句中挖掘出策略的主体属性、动作属性、客体属性以及属性间关系等属性信息，依据这些属性信息，即可直接得到可读、可执行的访问控制策略。

本文的主要贡献包括 3 个方面。1) 将基于属性的访问控制策略生成问题转化为自然语言处理问题，提出了一种基于深度学习的 ABAC 策略生成框架，该框架将策略生成问题分解为访问控制语句识别和访问控制属性挖掘 2 项任务，能够实现访问控制策略的自动化生成。2) 提出了一种基于混合神经网络架构的网络模型 BiGRU-CNN-Attention，实现

了访问控制策略语句识别，并在公开数据集中进行了测试，平均 F1-score 指标能够达到 0.941，比当前的 state-of-the-art 方法性能提高了 4.1%。3) 提出了一种基于双向长短期记忆网络（BiLSTM, bidirectional long short term memory）和条件随机场（CRF, conditional random field）的融合网络模型 AM-BiLSTM-CRF，实现了访问控制语句的属性挖掘，为访问控制策略生成提供了属性支持。通过实验验证了与基准方法相比，所提方法具有更好的性能。

2 相关工作

现有的围绕 ABAC 访问控制策略自动化生成技术的研究主要包括 2 个研究方向，具体如下。一个研究方向是在信息系统中已有的访问控制策略信息（用户−权限关系）的基础上，生成 ABAC 访问控制策略。围绕该问题，Xu 等[21]通过从给定的用户权限关系中遍历元组，使用选择的元组作为构建候选规则的种子，尝试通过用约束代替属性表达式中的连接来泛化每个候选规则，以此覆盖用户权限关系中的其他元组，实现 ABAC 策略的挖掘。Das 等[12]通过二进制矩阵形式表示现有的用户−权限关系，将策略生成问题转化为矩阵最小化问题，并提出了一种启发式求解方法。但是，由于矩阵过于稀疏，求解空间过大，存在求解效率较低的问题。为此，Cotrini 等[15]设计了一种新的子群发现算法，通过可靠性阈值来降低搜索空间，提高求解效率。Karimi 等[14]提出了一种基于无监督学习算法的策略生成方法，基于 K-modes 聚类算法实现近似策略规则模式的抽取，再从得到的模式中挖掘 ABAC 策略规则。但是该方法存在策略生成质量的稳定性不高，且难以设定恰当聚类值的问题。Mocanu 等[22]通过日志来训练一个受限的玻尔兹曼机（RBM, restricted Boltzmann machine）来提取策略规则。但在该研究中只给出了算法在策略空间中第一个阶段的初步结果，算法的最后一个阶段还并未实现。以上方法都只关注允许类型的访问控制策略，无法解决禁止类型策略的生成问题。针对此问题，Iyer 等[17]提出了一种基于子类枚举的算法，在牺牲一定计算效率的条件下，能够同时发现允许类型的授权规则和禁止类型的授权规则。

另一个研究方向是不需要借助信息系统中已有的访问控制策略信息（用户−权限关系）依据项目规范类文档生成访问控制策略。早期研究[23]是通过安全专家的人工分析来从文档中提取相应的访问控制策略，或者是在受控自然语言（CNL, controlled natural language）条件下[24-25]进行访问控制策略的提取。虽然人工分析通常能够生成最准确的结果，但代价是需要更熟练的安全专家和更多的评估时间。由于 CNL 被设计用来尽量减少自然语言中的模糊性和复杂性[25]，使用 CNL 可以产生较全面的结果，但是 CNL 通常需要专门的生成工具来对文档中的相关词汇进行转换。因此，基于 CNL 的方法灵活性较低，其应用场景较为受限。另外，由于实际环境中的绝大多数文档都是一般化的通用类别的自然语言文档。因此，与其他技术相比，自然语言处理技术（NLP, natural language processing）通常需要较少的人工工作，即可以处理通用类别的文档，在灵活性与可扩展性方面更具优势。

当前已存在一些使用自然语言处理技术从文档中自动提取访问控制策略的研究。Xiao 等[26]提出了一个名为 Text2Policy 的自动化方法，该方法根据 4 种预设的策略语义模式进行匹配，能够从包含访问控制策略的文档中提取出基于角色的访问控制策略。但是，Text2Policy 方法只适用于符合特定模式的需求规范，并依赖于匹配 4 个特定的句型来获取策略信息，无法捕捉到不遵循预先设定语义模式的策略。为了解决 Text2Policy 提取语义模式受限的问题，Slankas 等[27]提出了一种基于访问控制策略模式匹配的机器学习算法，该算法的核心是基于初始的种子模式生成适当的依赖图模式，并不断引导和提取新的模式来扩展原有的访问控制模式集，从而实现更好的策略提取效果。文献[28]提出了一种基于最小生成树（MST, minimum spanning tree）的迭代算法（ACRE, access control rule extraction）来提取非结构化文档中的访问控制语句。该算法将语句表示为以单词为顶点、单词间关系为边的解析图，通过在解析图中匹配访问控制策略匹配模式生成 MST 来提取策略。为了提高策略提取的准确率，还构造了一个朴素贝叶斯分类器来扩展策略匹配模式。

针对自然语言文档中的访问控制策略语句的识别问题，文献[29-30]基于语义角色标记（SRL, semantic role labeling）自动地识别谓词−参数结构（PAS, predicate-argument structure），然后对提取的参数使用一组预定义的规则来从自然语言需求文档中提取访问控制策略，并利用提取到的访问控制

策略来定义角色和构建 RBAC 系统。文献[19]设计了 Security features、PMI features、Syntactic complexity features 和 Dependency features 共 4 类特征来对文档中的语句进行描述，使用朴素贝叶斯分类器和支持向量机（SVM, support vector machine）实现访问控制语句识别。文献[31]采用递归神经网络模型（RNN, recurrent neural network）从自然语言文档中识别策略语句。总体来说，现有基于 NLP 的策略生成技术整体性能较一般。针对访问控制属性提取问题，Alohaly 等[32-33]使用卷积神经网络（CNN, convolutional neural network）从自然语言策略语句中识别出与系统访问控制相关的主体与客体属性集合。但是，该方法只是为了提取与访问控制相关的属性信息，从而为构建 ABAC 模型提供属性支撑，并没有获取不同属性之间的策略关系，无法依据这些属性来构建访问控制策略。

3　访问控制策略生成框架

3.1　基于属性的访问控制模型

ABAC 能够灵活利用实体所拥有的属性集来决定是否赋予其访问权限，具有较强的语义表达能力，适用于解决开放计算环境中细粒度访问控制和大规模动态授权问题，有助于实现高效的访问控制执行标准，缩短新应用服务的部署时间，被誉为“下一代”授权模型[34]，其核心概念如下。

定义 1　属性（attribute）用来描述参与到访问控制过程中实体的特征信息，由属性名与属性值构成，包括主体属性（S）、客体属性（O）、操作属性（A）和环境属性（E）。其中，主体属性描述访问请求发起方所具有的属性信息（如角色、单位等），客体属性描述能够被访问的资源所具有的属性信息（如名称、安全等级等），操作属性描述主体对客体的各种操作行为（如读取、写入等），环境属性描述访问控制过程中所受到的环境约束（如时间、空间等）。在访问控制策略中，主体属性、客体属性、操作属性是必须要素，环境属性是非必须要素。同时，环境属性包含了对访问控制的时空约束，较复杂，这部分内容将在下一步工作中进行详细研究。因此，为了简化问题，使研究更有针对性，本文只围绕包含主体属性、客体属性、操作属性这些必须要素的策略生成问题展开研究，在后文中对环境属性不再进行额外说明。

定义 2　属性元组（attribute tuple）是刻画访问控制实体特定类别属性的集合，是属性动态指派关系的体现，可表示为 X-tuple=$\{a_1, a_2, \cdots, a_n\}$，$X \in \{S, O, A\}$。

定义 3　访问控制策略（access control policy）是主体访问客体的规则和主体对客体授权逻辑的具体体现，可表示为四元组 ACP=(S-tuple, A-tuple, O-tuple, Sign)的形式，Sign∈{permit, deny}表示允许访问或禁止访问。

定义 4　访问请求（access request）是对资源的请求访问者、被访问的客体以及被请求操作的描述，可以表示为三元组 AR= (S-tuple, A-tuple, O-tuple)的形式。访问请求中至少包含一个主体属性、一个客体属性和一个操作属性。

定义 5　权限判决（permission decision）是在给定的访问控制策略评估环境中，针对用户的访问请求，做出用户允许或禁止访问相应资源的判决响应，可表示为一个映射函数 Decision: AR → {permit, deny}。

3.2　策略生命周期

图 1 是文献[34]给出的基于属性的访问控制策略的生命周期。在传统专家知识驱动的策略管理过程中，信息系统所有者负责定义访问控制保护用例，安全人员负责为给定的用例收集访问控制需求、定义访问控制属性、编写相应的访问控制策略，再由应用程序开发人员进行策略用例的测试、访问控制框架和访问控制策略的部署，最后由审计员负责进行 ABAC 的访问控制审计。本文的研究重点聚焦在收集访问控制需求（阶段②）、获取访问控制所需属性（阶段③）、编写访问控制策略（阶段④）这 3 个阶段。利用深度学习技术实现自动化、智能化的策略生成。

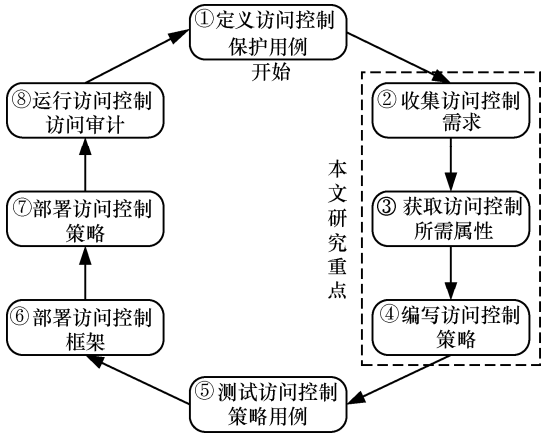


图 1　基于属性的访问控制策略的生命周期

3.3 策略生成框架

访问控制策略生成框架如图 2 所示。首先，对待处理的自然语言文档进行解析，在访问控制策略语句识别引擎中提取出包含访问控制信息的语句，这些语句描述了拥有哪些属性的主体能够以何种方式访问具有哪些属性的客体。一旦访问控制策略语句被提取出来之后，就对语句中所包含的主体属性、操作属性以及客体属性进行挖掘，生成相应策略元素。然后，直接将这些策略元素转化为可读、可执行的标准格式 ABAC 策略。再经过进一步的策略修正和验证步骤之后，将最终的 ABAC 策略存储到访问控制策略数据库中，完成从自然语言文本中提取访问控制策略的全部流程。下面，将对访问控制语句识别和访问控制属性挖掘这 2 项核心任务的解决方案进行详细说明。

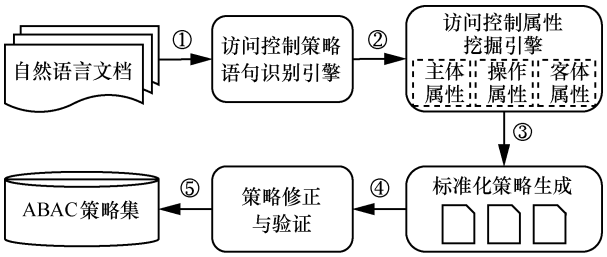


图 2 策略生成框架

4 访问控制策略语句识别引擎

本节提出了一种混合神经网络模型 BiGRU-CNN-Attention 来实现访问控制策略语句的识别。该模型由 Word embedding 层、隐含层和输出层 3 个部分组成。其中，隐含层包含 BiGRU 层、卷积层、池化与注意力层、合并层和全连接层，从而构成了一个如图 3 所示的 7 层神经网络结构。

4.1 Word embedding 层

Word embedding 层是访问控制语句识别模型的数据输入层，在该层中本文使用了谷歌最新提出的预训练 BERT（bidirectional encoder representation from transformer）模型[35]。BERT 模型将传统大量在下游具体 NLP 任务中做的操作转移到预训练的语言模型中，进一步增加了词向量模型的泛化能力，充分地对字符级、词级、句子级关系特征进行了描述。BERT 模型基于双向 transformer 技术进行词向量模型的训练，具有更深的层数和更好的并行性，在多项 NLP 任务中都具有非常优异的性能。BERT 模型的具体技术细节不作为本文研究的重点。本文基于 BERT 模型将自然语言文档中的词及其对应的特征进行编码，转化为词向量形式作为模型输入。

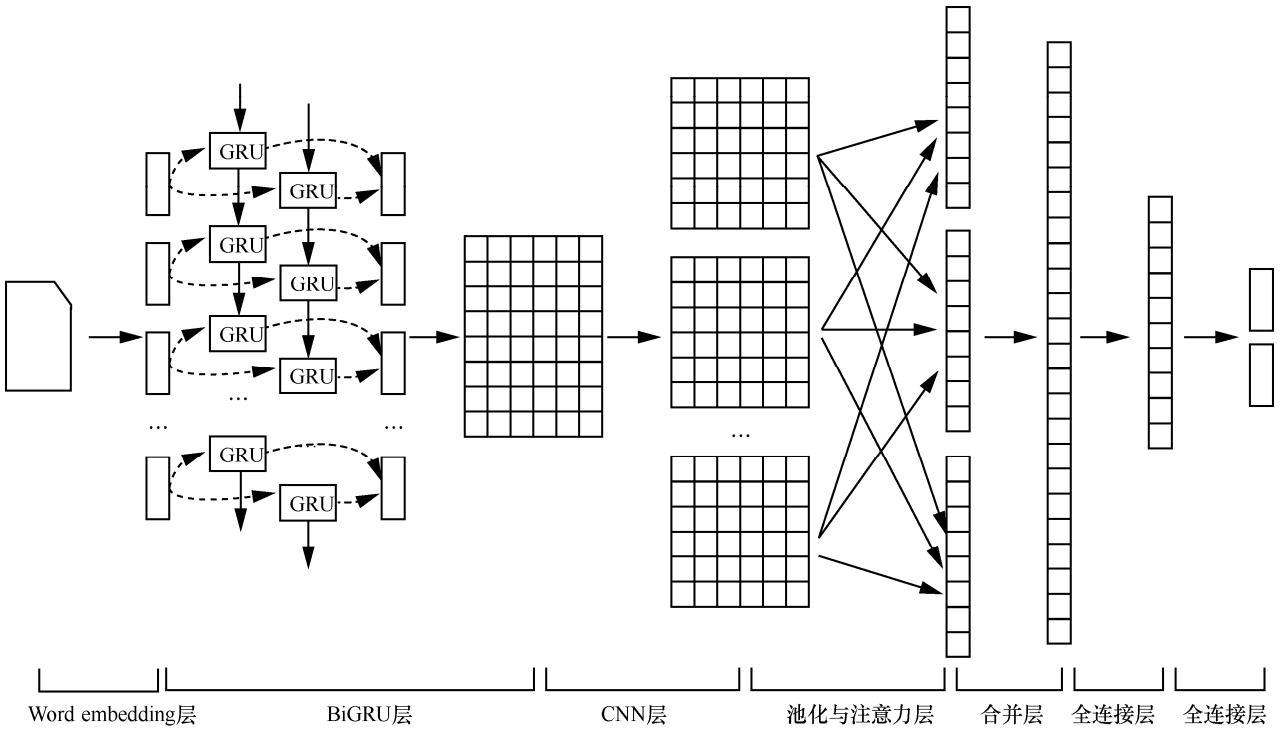


图 3 访问控制语句识别模型

4.2 BiGRU 层

门控循环单元（GRU, gated recurrent unit）是一种继承了长短期记忆网络（LSTM, long short-term memory）特性的神经网络结构，在某些应用场景下有近似 LSTM 的性能，但却具有更加简单的网络结构。当整体神经网络模型规模较大时，它拥有更少的参数和更好的收敛效果。双向门控循环单元（BiGRU, bidirectional gated recurrent unit）[36]由正反 2 个方向的 GRU 组成，相比单向 GRU 能够提取出更加全面的语句特征。因此，本文选取 BiGRU 来获取文本语句的深层次特征表示。

GRU 由更新门和重置门 2 个门组成。更新门用于控制前一时刻输出对后一时刻输出的影响程度，更新门的值越大，说明前一时刻输出对后一时刻输出的影响越大。重置门用于控制前一时刻输出被后一时刻忽略的程度，重置门的值越小，说明后一时刻忽略的信息越多。GRU 结构单元的更新方法为

$$\begin{aligned} z_{(t)} &= \sigma(W_z x_{(t)} + u_z h_{(t-1)}) \\ r_{(t)} &= \sigma(W_r x_{(t)} + u_r h_{(t-1)}) \\ \tilde{h}_{(t)} &= \tanh(W_h x_{(t)} + u_h (r_{(t)} \odot h_{(t-1)})) \\ h_{(t)} &= (1 - z_{(t)}) h_{(t-1)} + z_{(t)} \tilde{h}_{(t)} \end{aligned} \tag{1}$$

其中，$z_{(t)}$、$r_{(t)}$、$\tilde{h}_{(t)}$、$h_{(t)}$分别表示时刻 t 的更新门、重置门、候选激活状态、激活状态，$h_{(t-1)}$表示时刻 $t-1$ 的隐含层状态。由式(1)可知，$z_{(t)}$由当前时刻输入的信息与上一时刻需要被遗忘的信息共同决定，$r_{(t)}$由当前时刻输入的信息与上一时刻需要被继承的信息共同决定。

BiGRU 将 2 个方向相反的 GRU 输出进行合并，计算方法为

$$H_{(t)} = \boldsymbol{w}_{(t)} \boldsymbol{h}_{(t)} + \boldsymbol{w}'_{(t)} \boldsymbol{h}'_{(t)} + b_{(t)} \tag{2}$$

其中，$\boldsymbol{h}_{(t)}$ 和 $\boldsymbol{h}'_{(t)}$ 分别表示前向 GRU 和反向 GRU 中结构单元输出的隐含层向量，$\boldsymbol{w}_{(t)}$ 和 $\boldsymbol{w}'_{(t)}$ 分别表示 $\boldsymbol{h}_{(t)}$ 和 $\boldsymbol{h}'_{(t)}$ 对应的权重，$b_{(t)}$表示时刻 t 的偏置。

4.3 CNN 层

BiGRU 能够较好地提取出文本内双向时序维度的特征关系，但是由于文本中的词向量特征通常与其相临近的词向量特征之间也具有一定的语义关联，为了更好地对相邻特征间的关联进行语义分析，本文利用 CNN 层具有空间局部感知能力和权值共享网络结构的特点，来进一步地提取关联特征。并且，在保留数据主要特征的同时，有效地降低神经网络模型训练的复杂程度和参数数量。该方法能够有效避免过拟合，提高模型的泛化能力。输入是多个映射，输出是降维后的映射。每个映射都是属于上层的输入映射卷积值的组合，计算方法为

$$a_j^{(l)} = f(u_j^{(l)}) = f\left(\sum_{i \in N_j} a_j^{(l-1)} K_{i,j}^{(l)} + b_j^{(l)}\right) \tag{3}$$

其中，N_j 是输入映射的集合，$K_{i,j}^{(l)}$ 是用于连接第 i 个输入特征映射和第 j 个输出特征映射的卷积核，$b_j^{(l)}$ 是第 j 个特征映射的偏置项，f 是激活函数。

4.4 池化与注意力层

池化层也被称为下采样层，一般取池化区域中的最大值或平均值（分别称作最大池化或平均池化）。池化层能够减弱数据变形的影响，降低特征映射维度，提高模型的精度，避免过拟合的发生。本文为了提高模型的稳健性，同时采取了最大池化计算与平均池化计算，通过这种混合池化计算方法来降低单一池化可能会造成的数据方差增大与均值偏移的问题。计算方法为

$$\begin{aligned} &\text{max-pooling} = \text{downsample}\left(\max(s_{i,j})\right), i, j \in A_p \\ &F_1 = f\left(\beta_j^{(l)} \text{max-pooling}(a_j^{(l-1)} + b_j^{(l)})\right) \\ &\text{mean-pooling} = \text{downsample}\left(\text{mean}(s_{i,j})\right), i, j \in A_p \\ &F_2 = f\left(\beta_j^{(l)} \text{mean-pooling}(a_j^{(l-1)} + b_j^{(l)})\right) \end{aligned} \tag{4}$$

其中，s 表示池化区域 A_p 中所对应输入数据元素；max 表示对池化区域特征取最大化值；mean 表示对池化区块特征取平均值；表示下采样函数，包括最大池化 max-pooling 和平均池化 mean-pooling；表示偏置项；F_1 表示最大池化计算结果，F_2 表示平均池化计算结果。

起源于人类视觉注意力的注意力（Attention）机制在自然语言处理、图像处理、语音识别等领域表现出了非凡的性能。因此，本文引入 Attention 机制进行访问控制语句的识别。Attention 机制通过对数据进行加权处理，把不同的部分间数据联系起来，能够对语句中的重点词汇进行着重的关注与处理，从而提高系统的整体性能。使用 Attention 机制在输入语句中分配不同的关注度，突出局部的重要信息，从而使重要信息得到更多的关注。一般情况下，如果 BiGRU 和 CNN 得到的所有词向量在句子 S 中都被平等地处理，那么在一些不重要的词上将会浪费过多的计算时间。因此，本文通过对语句中的重点词进行着重关注，对输入序列中的每个元素

赋予权重，并将注意力集中在输入语句中最重要的信息部分，计算方法为

$$e_i = \tanh(w_i h_i + b_i)$$

$$\alpha_i = \exp(e_i)(\sum_{i=1}^{n}\exp(e_i))^{-1}$$

$$F_3 = \sum_{i=1}^{t}\alpha_i h_i \tag{5}$$

其中，α 是句子中新的隐含层的状态所占的权重，h_i 是向 Attention 机制中输入的初始隐含层状态，e_i 是时刻 i 隐含层状态的能量值，w_i 是权重系数，b_i 是对应时刻 i 的偏置。

4.5 合并层与全连接层

合并层接收来自池化与注意力层输出的最大池化计算结果 F_1、平均池化计算结果 F_2 以及注意力计算结果 F_3，将 3 个结果进行合并拼接之后，得到本层的融合特征 $a^{(l)}$，计算方法为

$$a^{(l)} = F_1 \oplus F_2 \oplus F_3 \tag{6}$$

其中，$\oplus$ 表示拼接操作。

然后，再将融合特征 FF 输入 2 个全连接层中，全连接层的计算与普通神经网络的计算一致，其输出为

$$a^{(l+1)} = f(w^{(l+1)}a^{(l)} + b^{(l+1)}) \tag{7}$$

最后，在最后一个全连接层中，对输出特征 f 进行 Softmax 函数计算，得到输入的文本语句为访问控制策略语句的概率。Softmax 函数的计算方法为

$$S_k(f) = \frac{\exp(f)}{\sum_K \exp(f)} \tag{8}$$

其中，K 为语句类别（属于访问控制语句为 1，非访问控制语句为 0），$S_k(f)$ 为相应的语句类别概率。

5 访问控制属性挖掘引擎

本节将属性挖掘问题转化为主体属性、客体属性以及动作属性的序列标注问题，提出的访问控制属性挖掘神经网络模型 AM-BiLSTM-CRF 如图 4 所示。AM-BiLSTM-CRF 网络模型是一个具有 CRF 的双向 LSTM 模型。首先，将访问控制语句中的单词进行向量化处理。本模型与 3.1 节类似，同样使用 BERT 模型预训练模型将访问控制语句中的词映射到高维向量空间，得到词向量 $\boldsymbol{W}$=[w_1, w_2, …, w_n]。然后，将 Word embedding 层中的词向量 $\boldsymbol{W}$ 输入前向 LSTM 和反向 LSTM 之间，依据上下文语义环境对特征进行学习，并将前向 LSTM 和反向 LSTM 的输出进行拼接得到 CRF 层的输入。最后，由 CRF 层学习不同词中属性标签间的依赖关系，生成面向访问控制语句的属性挖掘模型。

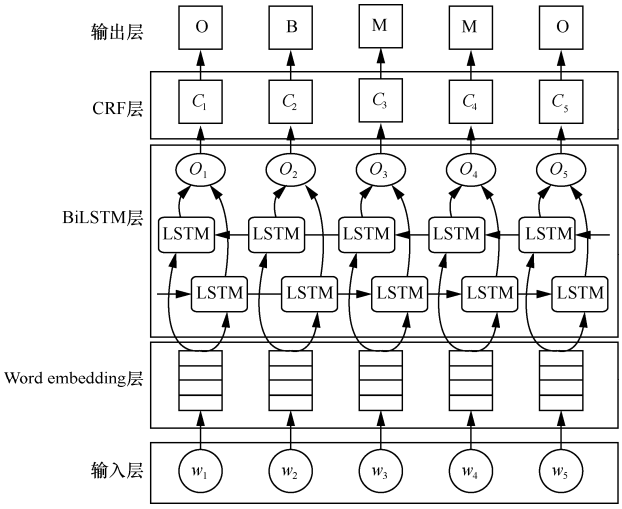


图 4 访问控制属性挖掘模型

5.1 标记方案

本文设计了 OBM 属性标记方案，对访问控制语句中的词进行属性标注，含义如下。标记符号 O 用于标注与访问控制无关的属性，标记符号 B 用于标注属性的起初位置，标记符号 M 用于标注属性的非起初位置。访问控制语句中共有主体属性、客体属性、动作属性 3 类属性需要进行标注，共包括 7 类标记，如表 1 所示。

表 1 属性标记方案

编号	标记符号	标记类别
1	/O	无关属性
2	/B_subject_attribute	主体属性起初位置
3	/M_subject_attribute	主体属性非起初位置
4	/B_action_attribute	动作属性起初位置
5	/M_action_attribute	动作属性非起初位置
6	/B_object_attribute	客体属性起初位置
7	/M_object_attribute	客体属性非起初位置

对于中文访问控制语句“注册的教授可以访问他的课程信息”的标注结果如下。

例 1 /B_subject_attribute: 注 /M_subject_attribute: 册 /M_subject_attribute: 的 /B_subject_attribute: 教 /M_subject_attribute: 授 /O: 可 /O: 以

/B_action_attribute:访/M_action_attribute:问/B_ object_attribute: 他 /M_object_attribute: 的 /B_object_ attribute:课/M_object_attribute:程/M_object_attribute: 信/M_object_attribute:息。

由于英文的表达形式与中文存在一定差异，因此标注情况略有不同，英文访问控制语句“A registered professor can access his course information”的标注结果如下。

例 2　/O:A /B_subject_attribute: registered /B_subject_attribute: professor /O: can /B_action_ attribute: access /B_object_attribute: his /B_object_ attribute: course /M_object_attribute: information。

5.2　BiLSTM 网络

LSTM 是一种特殊的 RNN 模型，能够解决传统循环神经网络中存在的上下文长期依赖问题，更适用于处理时序数据，其结构如图 5 所示。

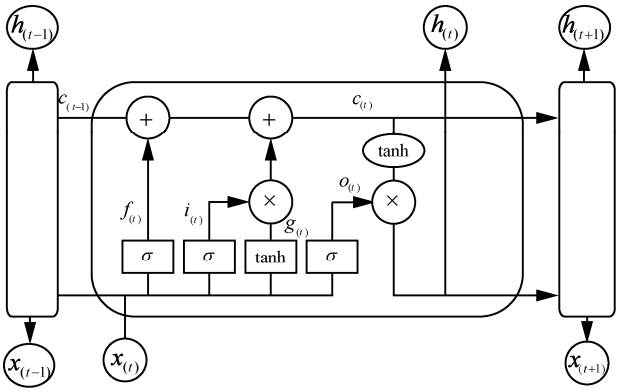


图 5　LSTM 结构

考虑到访问控制文本中的上下文词语存在相关性，一个词语可能与其前一个和下一个词语都存在相应关联。LSTM 只能利用历史的数据信息，无法利用数据中未来的数据信息。在这种情况下，使用 BiLSTM 将 2 个时序方向相反的 LSTM 连接到同一个网络输出中。通过这种结构，BiLSTM 增加了 LSTM 中的可计算信息，使网络模型既可以获取历史信息，也能够获取未来信息。BiLSTM 中包括输入门 i、遗忘门 f、输出门 o 和细胞状态 c 共 4 个部分，单个 LSTM 结构单元的更新为

$$\begin{aligned} i_{(t)} &= \sigma(\boldsymbol{W}_i \boldsymbol{x}_{(t)} + \boldsymbol{u}_i \boldsymbol{h}_{(t-1)} + \boldsymbol{b}_i) \\ f_{(t)} &= \sigma(\boldsymbol{W}_f \boldsymbol{x}_{(t)} + \boldsymbol{u}_f \boldsymbol{h}_{(t-1)} + \boldsymbol{b}_f) \\ o_{(t)} &= \sigma(\boldsymbol{W}_o \boldsymbol{x}_{(t)} + \boldsymbol{u}_o \boldsymbol{h}_{(t-1)} + \boldsymbol{b}_o) \\ g_{(t)} &= \tanh(\boldsymbol{W}_g \boldsymbol{x}_{(t)} + \boldsymbol{u}_g \boldsymbol{h}_{(t-1)} + \boldsymbol{b}_g) \end{aligned}$$

$$\begin{aligned} c_{(t)} &= f_{(t)} \otimes c_{(t-1)} + i_{(t)} \otimes g_{(t)} \\ \boldsymbol{h}_{(t)} &= o_t \otimes \tanh(c_t) \end{aligned} \tag{9}$$

其中，$i_{(t)}$、$f_{(t)}$、$o_{(t)}$、$c_{(t)}$分别表示在 t 时刻的输入门、遗忘门、输出门和细胞状态的值，$\boldsymbol{x}_{(t)}$表示 t 时刻的输入词向量，$\boldsymbol{h}_{(t)}$表示 t 时刻的隐含层向量，$C_t = \text{concat}(h_{\text{forward}}, h_{\text{backward}})$ 表示 sigmoid 激活函数，$\boldsymbol{W}$ 和 $\boldsymbol{u}$ 表示权重矩阵，$\boldsymbol{b}$ 表示偏置向量，$\boldsymbol{h}_{\text{forward}}$和 $\boldsymbol{h}_{\text{backward}}$ 分别表示 BiLSTM 中前向 LSTM 和反向 LSTM 中结构单元输出的隐含层向量。将 $\boldsymbol{h}_{\text{forward}}$和 $\boldsymbol{h}_{\text{backward}}$连接，得到 BiLSTM 在 t 时刻的输出为

$$C_t = \text{concat}(\boldsymbol{h}_{\text{forward}}, \boldsymbol{h}_{\text{backward}}) \tag{10}$$

其中，$\boldsymbol{h}_{\text{forward}}$ 和 $\boldsymbol{h}_{\text{backward}}$ 分别对应访问控制语句 2 个方向上的上下文信息。

5.3　CRF 网络

在属性挖掘过程中，当前词的属性标签通常与其周围词的属性标签是相关联的，例如属性标签 E 必须出现在属性标签 B 之后（属性标签的标记方案已在 4.1 节中详细说明）。CRF 通过计算相邻标签间的转移矩阵来得到一个属性标签在一个访问控制语句序列中转移到另一个属性标签的条件概率。从而，通过对转移矩阵的训练能够实现对属性标签之间依赖关系的学习。通过引入 CRF 层，将使属性挖掘的计算结果更准确。对于式(11)所示的给定的访问控制语句

$$\text{ACP} = (a_1, a_2, \cdots, a_n) \tag{11}$$

对应的属性标签预测结果序列为

$$\text{A\_tag} = (t_1, t_2, \cdots, t_n) \tag{12}$$

属性标签预测结果的评估分数为

$$\text{score}(\text{ACP}, \text{A\_tag}) = \sum_{i=0}^{n} T_{t_i, t_{i+1}} + \sum_{i=1}^{n} C_{i, t_i} \tag{13}$$

其中，$\boldsymbol{T}$ 表示属性预测标签的转移概率矩阵，其维度是 $(k+2)\times(k+2)$；$T_{n,m}$ 表示属性标签 n 与属性标签 m 间的转移概率得分；k 表示不同类别属性标签的数目；t_0=START 与 t_{n+1}=END 分别表示访问控制语句的起始标签与终止标签；$\boldsymbol{C}$ 表示 BiLSTM 网络的输出矩阵，其维度是 $n\times k$；C_{ij} 表示第 i 个词被预测为第 j 个属性标签的得分。对 ACP 语句的属性标签进行预测时，使用柔性最大值计算方法（softmax）对结果进行归一化处理，计算方法为

$$p(\text{A\_tag} \mid \text{ACP}) = \frac{\mathrm{e}^{\text{score}(\text{ACP}, \text{A\_tag})}}{\sum_{\bar{t} \in \text{TAG}_s} \mathrm{e}^{\text{score}(\text{ACP}, \bar{t})}} \tag{14}$$

其中，TAG 表示 ACP 语句中所有可能的属性标签序列。在属性挖掘的训练过程中，需要最大化预测结果为正确的属性标签序列的似然概率，计算方法为

$$\log(p(\text{A\_tag}|\text{ACP})) = \text{score}(\text{ACP},\text{A\_tag}) - \log\left(\sum_{\bar{t}\in\text{TAG}_s} \text{e}^{\text{score}(\text{ACP},\bar{t})}\right) \tag{15}$$

最后，在模型输出端将预测得分最高的属性标签序列作为最终的属性标签序列输出，即

$$\text{A\_tag} = \arg\max_{\bar{t}\in\text{TAG}_s}(\text{score}(\text{ACP},\bar{t})) \tag{16}$$

6 实验评估

6.1 实验设置与数据集

为了对本文所提方法的性能进行评估，本节在表 2 所列出的公开数据集[31]条件下进行实验，该数据集包括 iTrust、IBM App、Cyberchair、Collected ACP 共 4 类数据集，包括 2 477 条文本数据。这些数据集由 Slankas 等[28]通过人工标注得到。其中，iTrust 是一个以病人为中心的应用程序，用于维护电子健康记录；IBM App 是一款高校使用的课程管理系统；Cyberchair 是一个会议管理系统；Collected ACP 是由 Xiao 等[26]收集的访问控制策略语句组合而成的数据集。由于单一数据集的数据量有限，本节将 4 类数据集的数据汇总进行实验，并且按 70%、15%、15%的比例将数据集划分为训练集、验证集和测试集。同时，为了尽可能地降低数据随机性对实验结果的带来的影响，本文在已标注的数据集上采用 5 折交叉验证进行多次实验。实验的软硬件环境如下。操作系统为 Windows 10 64 位，CPU 为 Intel(R) Core(TM) i7- 4710MQ@ 2.5 GHz，GPU 为 GeForce GTX 850M，内存大小为 16 GB，Tensorflow 版本为 1.14.0，Keras 版本为 2.1.3，Python 版本为 3.6。

表 2 数据集描述

数据集	领域	ACP 语句数目/条	Non-ACP 语句数目/条	总数/条
iTrust	Healthcare	967	664	1 631
IBM App	Education	169	232	401
Cyberchair	Conference	140	163	303
Collected ACP	Multiple	125	17	142
总数	—	1 401	1 076	2 477

6.2 评估指标

本文使用准确率（Accuracy）、精确率（Precision）、召回率（Recall）和 F1 值（F1-score）作为实验性能的评估指标。Accuracy 表示文本识别结果是正确的样本数占样本总数的比例。Precision 表示被正确识别为访问控制策略语句的样本数占被识别为访问控制策略语句的样本数的比例。Recall 表示被正确识别为访问控制策略语句的样本数占真实情况为访问控制策略语句的样本数的比例，是覆盖范围的度量。F1-score 表示 Precision 和 Recall 的加权调和平均值。为了计算这些评估指标，分类器的预测结果被分为以下 4 类。TP（true positive）是被正确识别为访问控制策略语句的样本数，TN（true negative）是被正确识别为非访问控制策略语句的样本数，FP（false positive）是被错误识别为访问控制策略语句的样本数，FN（false negative）是被错误识别为非访问控制策略语句的样本数。评价指标对应的计算式分别为

$$\text{Accuracy}=\frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}}$$

$$\text{Precision}=\frac{\text{TP}}{\text{TP}+\text{FP}}$$

$$\text{Recall}=\frac{\text{TP}}{\text{TP}+\text{FN}}$$

$$\text{F1-score}=\frac{2\text{Precision}\cdot\text{Recall}}{\text{Precision}+\text{Recall}} \tag{17}$$

6.3 实验结果与分析

为了评估所提方法的性能，本文分别进行了 2 组实验对访问控制语句识别性能与访问控制属性挖掘性能进行评估。

6.3.1 访问控制语句识别性能评估

本实验的超参数设置如下。输入层使用 BERT 模型预训练语言模型将文本转化成词向量，在输入层之后加一个 rate 为 0.2 的 SpatialDropout1D 层，用于提高模型的活化能力。BiGRU 中隐含层的结构单元数量为 100，Conv1D 层中包括 128 个卷积核、卷积核尺寸为 2、步长为 1、激活函数为 ReLU，池化操作分别使用最大池化 GlobalMaxPooling1D 和平均池化 GlobalAveragePooling1D 来进行计算，在全连接层前加一个 rate 为 0.5 的 Droupout 层，避免模型过拟合。2 个全连接层使用的激活函数分别是 ReLU 函数和 Softmax 函数，训练过程中设置 batch_size 为 125，epoch 为 12，选取 Adam 优化器

来训练神经网络。

1) 不同神经网络模型识别准确率和 Loss 值评估

为了比较不同神经网络模型在访问控制语句识别任务的性能，本文选取了 4 个常用的神经网络模型作为基准对比模型，基准对比模型描述如下。

① CNN_LSTM 模型。先添加一个 CNN，再添加一个 LSTM 网络。

② BiLSTM 模型。单一 BiLSTM 网络。

③ CNN_GRU 模型。先添加一个 CNN，再添加一个 GRU 网络。

④ BiGRU 模型。单一 BiGRU 网络。

所有网络模型均采用 BERT 模型预训练语言模型作为词向量的输入，实验结果分别如图 6 和图 7 所示。与其他网络模型相比，在验证集中本文所提方法 BiGRU-CNN-Attention 虽然存在一定程度的波动，但总体的性能是最优的，能够达到最高 95.97% 的准确率和最低 0.177 2 的 Loss 值，基本能够满足真实环境下访问控制策略语句识别的性能要求。

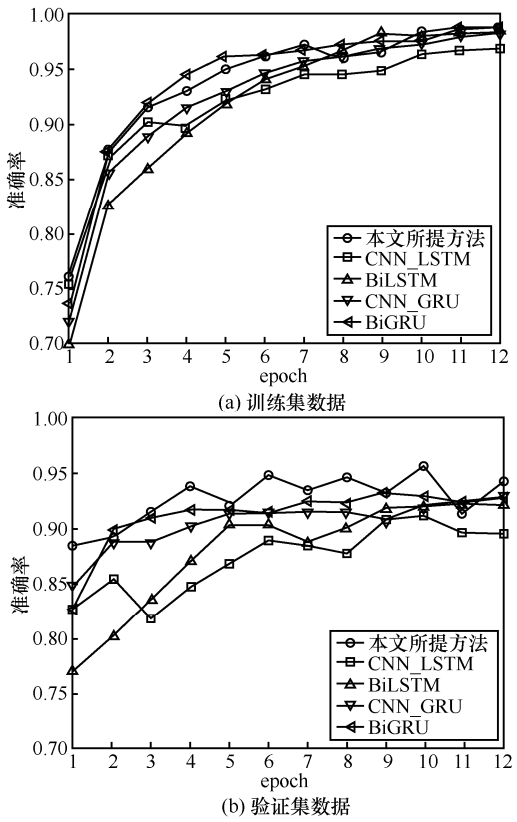


图 6　不同模型准确率随 epoch 的变化

2) 与现有基准 ACP 识别方法的比较

表 3 为与现有访问控制语句识别方法在精确率、召回率和 F1-score 上进行的对比。由实验结果可知，本文所提方法在 3 项指标上均为最优。平均 F1-score 指标能够达到 0.941，比当前的 state-of-the-art 方法性能提高了 4.1%。

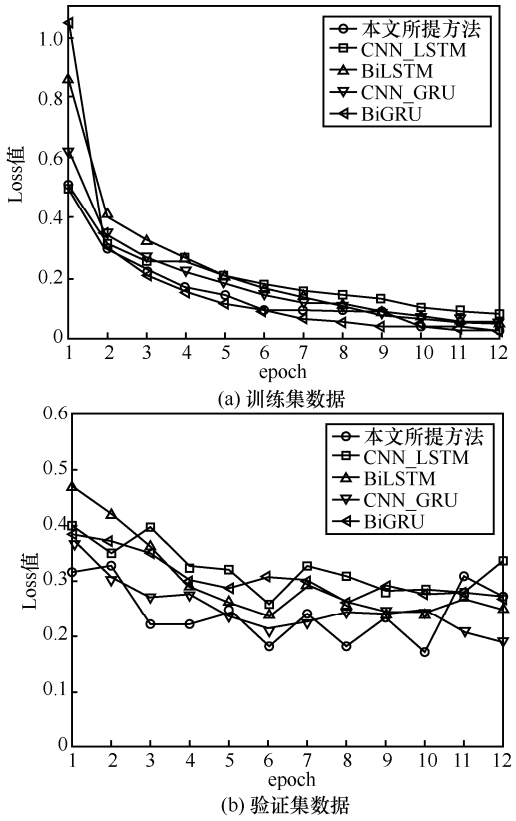


图 7　不同模型 Loss 值随 epoch 的变化

6.3.2　访问控制属性挖掘性能评估

为了增加对中文访问控制语句的访问控制属性挖掘性能进行评估，本文将表 2 数据集中的访问控制语句进行翻译，得到了对应的中文数据集。本实验的超参数设置如下。BiLSTM 中隐含层的结构单元数量为 128，训练过程中设置 batch_size 为 125，epoch 为 12，选择 Adam 优化器来训练神经网络。

1) 不同基准神经网络模型性能的对比

如图 8 和图 9 所示，在训练集和验证集的结果中，AM-BiLSTM-CRF 模型性能最优，在英文数据集和中文数据集中分别能够达到最高 95.41%和 96.88%的准确率。BiLSTM 模型的性能居中，CNN_LSTM 模型的性能最差。另外，从表 4 和表 5 中实验结果可知，在英文实验数据集中，本文所提方法在主体属性、动作属性、客体属性的性能上均达到最优。在中文实验数据集中，局部性能虽然不都是最优，但是整体的性能是较好的，原因如下。

表 3 **ACP 句子识别性能对比**

方法	数据集	性能		
		Precision	Recall	F1-score
文献[26]	iTrust、IBM App	0.887	0.894	0.891
文献[27]	iTrust	0.873	0.908	0.890
文献[28]	iTrust、IBM App、Cyberchair、Collected ACP	0.830	0.874	0.852
文献[19]	iTrust、IBM App、Cyberchair、Collected ACP	0.900	0.900	0.900
文献[31]	iTrust、IBM App、Cyberchair、Collected ACP	0.813	0.742	0.775
文献[30]	iTrust、IBM App、Cyberchair	0.583	0.863	0.657
文献[29]	iTrust、IBM App、Cyberchair	0.635	0.863	0.698
本文所提方法	iTrust、IBM App、Cyberchair、Collected ACP	**0.942**	**0.941**	**0.941**

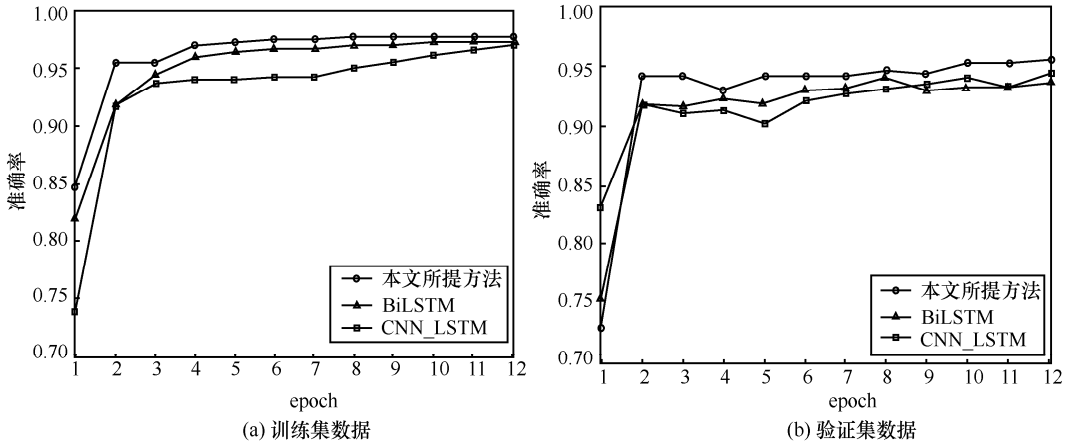


图 8 不同模型在英文数据集中准确率随 epoch 的变化

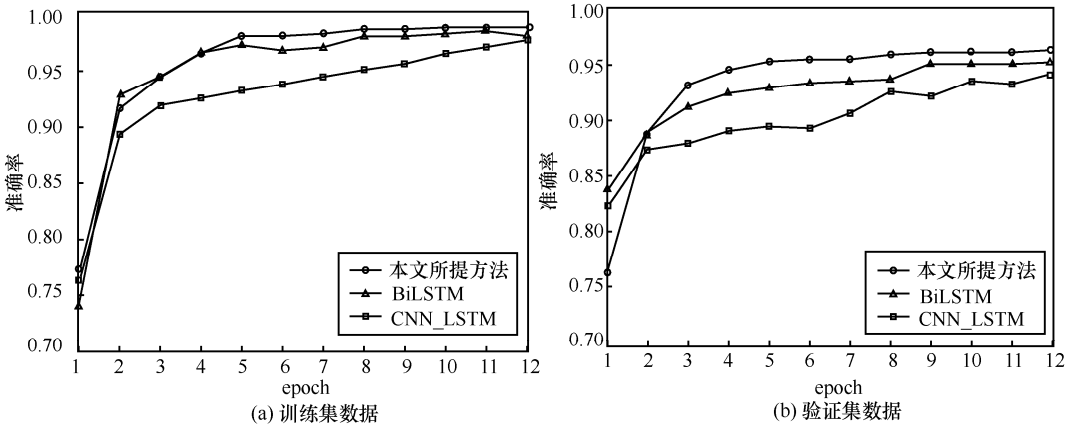


图 9 不同模型在中文数据集中准确率随 epoch 的变化

与 CNN_LSTM 模型相比，BiLSTM 能够从正向和反向这 2 个方向上同时对访问控制策略的属性特征进行学习，比单方向学习能够更好地利用文本内上下文的约束信息。与 BiLSTM 模型相比，本文通过引入 CRF 模型能够提升性能，这是因为访问控制属性信息通常为连续的文本片段，文本内相邻词间具有更强的依赖关系，CRF 模型能够通过转移概率的计算更好地捕捉到相邻文本元素标签之间的依赖关系，弥补单一 BiLSTM 模型所存在的相信标签关联能力不足的问题，从而进一步提高系统的性能。

2) 不同标记方案对系统性能的影响

除了 4.1 节中叙述的本文所使用的 OBM 标记方案，本文还尝试使用 OB 标记方案来对文本属性

表 4　　**英文数据集属性挖掘性能对比**

方法	主体属性			动作属性			客体属性		
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
BiLSTM	0.783 3	0.783 3	0.783 3	0.843 6	0.804 2	0.823 4	0.706 7	0.868 9	0.779 4
CNN_LSTM	0.717 0	0.666 7	0.690 9	0.830 3	0.830 1	0.827 2	0.626 5	0.787 9	0.698 0
本文所提方法	**0.785 7**	**0.830 2**	**0.807 3**	**0.867 6**	**0.881 4**	**0.873 0**	**0.791 0**	**0.898 3**	**0.841 3**

表 5　　**中文数据集属性挖掘性能对比**

方法	主体属性			动作属性			客体属性		
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
BiLSTM	0.875 5	**0.896 5**	**0.885 8**	0.850 7	0.866 4	0.858 4	0.786 2	**0.844 0**	0.814 1
CNN_LSTM	0.804 6	0.768 7	0.786 2	**0.972 4**	0.520 3	0.661 2	0.709 1	0.799 4	0.748 6
本文所提方法	**0.902 4**	0.868 0	0.884 8	0.885 0	**0.928 7**	**0.906 4**	**0.818 2**	0.831 7	**0.824 9**

进行标记，从而对比不同标记方案对系统性能的影响。其中，O 标记无关属性，B 标记相关属性。由表 6 和表 7 可知，本文所采取的 OBM 标记方案在不同的数据集中各项性能更优。

7　结束语

为了实现基于属性的访问控制策略的自动化生成，本文提出了一种基于深度学习的 ABAC 访问控制策略生成技术，为访问控制系统初始访问控制策略的生成提供了一条新的解决思路。首先，提出了一种基于深度学习的策略生成框架，该框架将策略生成问题分解为访问控制策略语句识别和访问控制属性挖掘两项核心任务。然后，分别设计了 BiGRU-CNN-Attention 和 AM-BiLSTM-CRF 这 2 个神经网络模型来解决访问控制策略语句识别任务和访问控制属性挖掘任务，从而为生成可读、可执行的访问控制策略提供支撑。实验结果验证了本文所提方法的有效性。下一步的工作将尝试进一步提高策略生成的性能，并对策略用例的自动化测试以及策略中环境属性的挖掘工作展开研究。

参考文献：

[1] 冯登国, 张敏, 李昊. 大数据安全与隐私保护[J]. 计算机学报, 2014, 37(1): 246-258.
FENG D G, ZHANG M, LI H. Big data security and privacy protection[J]. Chinese Journal of Computers, 2014, 37(1): 246-258.

[2] 房梁, 殷丽华, 郭云川, 等. 基于属性的访问控制关键技术研究综述[J]. 计算机学报, 2017, 40(7): 1680-1698.
FANG L, YIN L H, GUO Y C, et al. A survey of key technologies in attribute-based access control scheme[J]. Chinese Journal of Computers, 2017, 40(7): 1680-1698.

[3] SERVOS D, OSBORN S L. Current research and open problems in attribute-based access control[J]. ACM Computing Surveys, 2017, 49(4): 1-45.

[4] XIN J, KRISHNAN R, SANDHU R. A unified attribute-based access control model covering DAC, MAC and RBAC[C]//Proceedings of the 26th Annual IFIP WG 11.3 Conference on Data and Applications Security and Privacy. Berlin: Springer, 2012: 41-55.

[5] HU V, KUHN D, FERRAIOLO D. Attribute-based access control[J]. Computer, 2015, 48(2): 85-88.

[6] BUI T, STOLLER S D, LI J. Greedy and evolutionary algorithms for mining relationship-based access control policies[J]. Computers &

表 6　　**英文数据集下不同标记方案的性能对比**

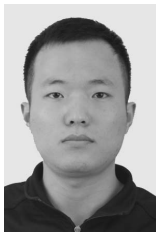
方法	主体属性			动作属性			客体属性		
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
OB	0.723 2	0.800 7	0.760 0	0.835 5	0.811 0	0.823 0	0.696 0	0.882 4	0.778 1
OBM	**0.785 7**	**0.830 2**	**0.807 3**	**0.867 6**	**0.881 4**	**0.873 0**	**0.791 0**	**0.898 3**	**0.841 3**

表 7　　**中文数据集下不同标记方案的性能对比**

方法	主体属性			动作属性			客体属性		
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
OB	0.805 5	0.777 0	0.791 0	0.802 0	0.831 0	0.816 2	0.699 0	0.844 8	0.765 0
OBM	**0.902 4**	**0.868 0**	**0.884 8**	**0.885 0**	**0.928 7**	**0.906 4**	**0.818 2**	0.831 7	**0.824 9**

Security, 2019, 80: 317-333.
[7] SANDERS M W, YUE C. Mining least privilege attribute based access control policies[C]//Annual Computer Security Applications Conference. New York: ACM Press, 2019: 404-416.
[8] VAIDYA J, ATLURI V, WARNER J, et al. Role engineering via prioritized subset enumeration[J]. IEEE Transactions on Dependable & Secure Computing, 2010, 7(3): 300-314.
[9] BAUMGRASS A, STREMBECK M. Bridging the gap between role mining and role engineering via migration guides[J]. Information Security Technical Report, 2013, 17(4): 148-172.
[10] HARIKA P, NAGAJYOTHI M, JOHN J C, et al. Meeting cardinality constraints in role mining[J]. IEEE Transactions on Dependable & Secure Computing, 2015, 12(1): 71-84.
[11] NEUMANN G, STREMBECK M. A scenario-driven role engineering process for functional RBAC roles[C]//Symposium on Access Control Models and Technologies. New York: ACM Press, 2002: 33-42.
[12] DAS S, SURAL S, VAIDYA J, et al. VisMAP: visual mining of attribute-based access control policies[C]//International Conference on Information Systems Security. Berlin: Springer, 2019: 79-98.
[13] TALUKDAR T, BATRA G, VAIDYA J, et al. Efficient bottom-up mining of attribute based access control policies[C]//IEEE 3rd International Conference on Collaboration and Internet Computing. Piscataway: IEEE Press, 2017: 339-348.
[14] KARIMI L, JOSHI J. An unsupervised learning based approach for mining attribute based access control policies[C]//International Conference on Big Data. Piscataway: IEEE Press, 2018: 1427-1436.
[15] COTRINI C, WEGHORN T, BASIN D. Mining ABAC rules from sparse logs[C]//IEEE European Symposium on Security and Privacy. Piscataway: IEEE Press, 2018: 31-46.
[16] GAUTAM M, JHA S, SURAL S, et al. Poster: constrained policy mining in attribute based access control[C]//Symposium on Access Control Models and Technologies. New York: ACM Press, 2017: 121-123.
[17] IYER P, MASOUMZADEH A. Mining positive and negative attribute-based access control policy rules[C]/Symposium on Access Control Models and technologies. New York: ACM Press, 2018: 161-172.
[18] KUHLMANN M, SHOHAT D, SCHIMPF G, et al. Role mining-revealing business roles for security administration using data mining technology[C]//Symposium on Access Control Models and Technologies. New York: ACM Press, 2003: 179-186.
[19] NAROUEI M, KHANPOUR H, TAKABI H. Identification of access control policy sentences from natural language policy documents[C]//IFIP Annual Conference on Data and Applications Security and Privacy. Berlin: Springer, 2017: 82-100.
[20] NAROUEI M, TAKABI H, NIELSEN R D. Automatic extraction of access control policies from natural language documents[J]. IEEE Transactions on Dependable and Secure Computing, 2018, 17(3): 1.
[21] XU Z, STOLLER S D. Mining attribute-based access control policies[J]. IEEE Transactions on Dependable and Secure Computing, 2015, 12(5): 533-545.
[22] MOCANU D C, TURKMEN F, LIOTTA A. Towards ABAC policy mining from logs with deep learning[C]//In Proceedings of International Multi Conference. Piscataway: IEEE Press, 2015: 10-16.
[23] HE Q, ANTÓN A I. Requirements-based access control analysis and policy specification (ReCAPS)[J]. Information & Software Technology, 2009, 51(6): 993-1009.
[24] SHI L L, CHADWICK D W. A controlled natural language interface for authoring access control policies[C]//Applied Computing. New York: ACM Press, 2011: 1524-1530.
[25] SCHWITTER R. Controlled natural languages for knowledge representation[C]//International Conference on Computational Linguistics. New York: ACM Press, 2010: 1113-1121.
[26] XIAO X, PARADKAR A, THUMMALAPENTA S, et al. Automated extraction of security policies from natural-language software documents[C]//ACM Sigsoft International Symposium on the Foundations of Software Engineering. New York: ACM Press, 2012: 1-11.
[27] SLANKAS J, WILLIAMS L. Access control policy extraction from unconstrained natural language text[C]//2013 International Conference on Social Computing. New York: ACM Press, 2013: 435-440.
[28] SLANKAS J, XIAO X, WILLIAMS L, et al. Relation extraction for inferring access control rules from natural language artifacts[C]//Proceedings of the 30th Annual Computer Security Applications Conference. New York: ACM Press, 2014: 366-375.
[29] NAROUEI M, TAKABI H. Automatic top-down role engineering framework using natural language processing techniques[C]//International Conference Information Security Theory and Practice. New York: ACM Press, 2015: 137-152.
[30] NAROUEI M, TAKABI H. Towards an automatic top-down role engineering approach using natural language processing techniques[C]//Symposium on Access Control Models and Technologies. New York: ACM Press, 2015: 157-160.
[31] NAROUEI M, KHANPOUR H, TAKABI H, et al. Towards a top-down policy engineering framework for attribute-based access control[C]//Symposium on Access Control Models and Technologies. New York: ACM Press, 2017: 103-114.
[32] ALOHALY M, TAKABI H, BLANCO E, et al. A deep learning approach for extracting attributes of ABAC policies[C]//Symposium on Access Control models and Technologies. New York: ACM Press, 2018: 137-148.
[33] ALOHALY M, TAKABI H, BLANCO E. Automated extraction of attributes from natural language attribute-based access control (ABAC) policies[J]. Cybersecurity, 2019, 2(1): 2-12.
[34] BROSSARD D, GEBEL G, BERG M. A systematic approach to implementing ABAC[C]//Proceedings of the 2nd ACM Workshop on Attribute-Based Access Control. New York: ACM Press, 2017: 53-59.
[35] DEVLIN J, CHANG M, LEE K, et al. BERT 模型: pre-training of deep bidirectional transformers for language understanding[C]//North American Chapter of the Association for Computational Linguistics. Virginia: NAACL, 2019: 4171-4186.
[36] LUO X, ZHOU W, WANG W, et al. Attention-based relation extraction with bidirectional gated recurrent unit and highway network in the analysis of geological data[J]. IEEE Access, 2018, 6: 5705-5715.

[作者简介]



刘敖迪（1992− ），男，黑龙江伊春人，信息工程大学博士生，主要研究方向为大数据安全、访问控制技术。

杜学绘（1968− ），女，河南辉县人，博士，信息工程大学教授、博士生导师，主要研究方向为网络信息安全。

王娜（1980− ），女，河南济源人，博士，信息工程大学副教授、硕士生导师，主要研究方向为大数据安全。

乔蕊（1983− ），女，河南周口人，博士，周口师范学院教授，主要研究方向为区块链安全。