DOI 10.36622/1682-7813.2024.27.2.009
УДК 004.056

МОДУЛЬ НЕЙРОСЕТЕВОЙ РЕГЛАМЕНТАЦИИ МЕР ПРОТИВОДЕЙСТВИЯ КИБЕРАТАКАМ

Г.А. Остапенко, А.П. Васильченко, А.А. Остапенко, А.А. Ноздрюхин, Д.С. Покудин, Н.Н. Корвяков

В статье рассматриваются особенности построения и функционирования нейросетевого модуля регламентации мер противодействия сетевым вторжениям. Описана архитектура данного модуля, включая его взаимодействие с модулем обнаружения и идентификации сетевых вторжений. Представлена инструкция по развертыванию локального варианта нейросети и сформирована структурная схема web-приложения, позволяющего администратору безопасности получать интеллектуальные подсказки в рамках регламентации мер противодействия.

Ключевые слова: нейросеть, языковая модель, меры противодействия, регламентирование мер противодействия, безопасность

Введение

В эпоху цифровой трансформации обеспечение информационной безопасности (ИБ) становится критически важным аспектом деятельности любой организации. В связи с постоянным совершенствованием технологий реализации кибернетических атак, нарушающих функционирование автоматизированных информационных систем (АИС), механизмы защиты должны быть подвержены постоянной модернизации с учетом современных трендов в области кибербезопасности. Одной из наиболее заметных тенденций в рамках ИТ-сферы стала интеграция нейросетевых технологий в различные информационные системы и процессы. Такой тренд связан с появлением крупных языковых моделей, таких как GPT-3.5, GPT-4, Claude-3, Mistral и др., которые помимо возможности задать вопрос в многопоточном чате предоставляют функционал API, позволяющий удобно использовать данные технологии в рамках собственных решений. При этом свое применение нейросетевые технологии находят и в области информационной безопасности. Одним из вариантов их использования является внедрение в процесс регламентации мер противодействия сетевым вторжениям для выдачи интеллектуальных подсказок. Основное преимущество внедрения нейросетевых технологий в процесс выдачи мер противодействия состоит в их способности к самосовершенствованию, самообучению, что позволяет им эффективно распознавать и предотвращать как известные, так и новые виды кибератак. В зависимости от количества данных, на основании которых проводилось машинное обучение языковой модели, точность выдаваемых ответов у нейросетей может достигать 80%. Однако данные исследования проводились на датасетах, затрагивающих сразу несколько предметных областей [1]. Рассматривая использование языковой модели в контексте одной предметной области с заранее проведенным дообучением под ее специфику, точность выдаваемых ответов можно повысить, что в свою очередь отразится и на качестве выдачи интеллектуальных подсказок, являющихся основной задачей нейросетевого модуля регламентации противодействия кибератакам. В связи с вышеуказанными тезисами девизом подготовки современных специалистов по защите информации можно сделать перефразированный афоризм: «Ученым можешь ты не быть, но нейросеть познать обязан», ибо сегодня каждый выпускник должен быть способен пользоваться нейросетевыми инструментами.

Таким образом, модуль нейросетевой регламентации противодействия кибератакам должен представлять собой интеллектуальную систему, способную

адаптироваться к меняющимся угрозам и обеспечивать эффективную защиту информационных ресурсов. Цель его реализации заключается в повышении защищенности АИС за счет внедрения нейросетевых технологий в процесс регламентации мер противодействия сетевым вторжениям. При этом актуальность состоит в использовании высоко эффективного инструмента, позволяющего удовлетворять растущие требования к безопасности информационных систем и сетей в условиях постоянно меняющегося ландшафта киберугроз. Кроме того, при проектировании данного модуля обучающийся приобретает необходимые компетенции:

– агрегация, генерация и форматирование баз профессиональных знаний и данных под нейросетевую реализацию;

– организация машинного обучения нейросети этими сведениями;

– развертывания локальной языковой модели на сервере и обращении к ней посредством API;

– подготовка запросов и использование интеллектуальных подсказок в проектных ситуациях защиты информации.

Архитектура автоматизированной системы обнаружения и идентификация сетевых вторжений с регламентацией мер противодействия

Модуль регламентирования мер противодействия (МРМП) является одной из основных частей общей автоматизированной системы обнаружения и идентификации сетевых вторжений с регламентацией мер противодействия. МРМП должен быть развернут на компьютере администратора безопасности, который обозначен как ЦОД, причем предварительно должно быть настроено взаимодействие с компьютерами из локальной сети АИС. К его основным функциям относятся:

– определение списка ip-адресов компьютеров в локальной сети АИС (это необходимо для обеспечения удобства выдачи интеллектуальных подсказок для конкретного хоста);

– прием входных данных от модуля обнаружения и идентификации сетевых вторжений (МОиИСВ) в формате UDP-пакета, содержащего данные об атаке и уязвимостях в текстовой форме, это показано на рис.1;

– выдача мер противодействия сетевой атаке с учетом имеющихся уязвимостей по пришествию сигнала о наличии атаки на хосте от МОиИСВ;

– обеспечение возможности приема запросов от администратора безопасности и выдачи интеллектуальных подсказок конкретно по каждому хосту с сохранением истории чата.

Рис. 1. Структура UDP-пакета, приходящего в модуль регламентирования мер противодействия

Необходимо отметить, что в структуре UDP-пакета присутствует разделитель «|», как механизм для преобразования данных в массив, содержащий текстовое представление сведений об атаке и уязвимостях. Так как данные в UDP-пакете хранятся в численном виде, то при отсутствии разделителей обеспечить преобразование, при котором будет видно, где заканчивается название атаки и начинается наименование новой уязвимости, не представляется возможным в виду того, что вся информация будет выглядеть как сплошной текст без пробелов и других знаков, благодаря которым можно было бы произвести разделение. Архитектура МРМП отображена на рис. 2.

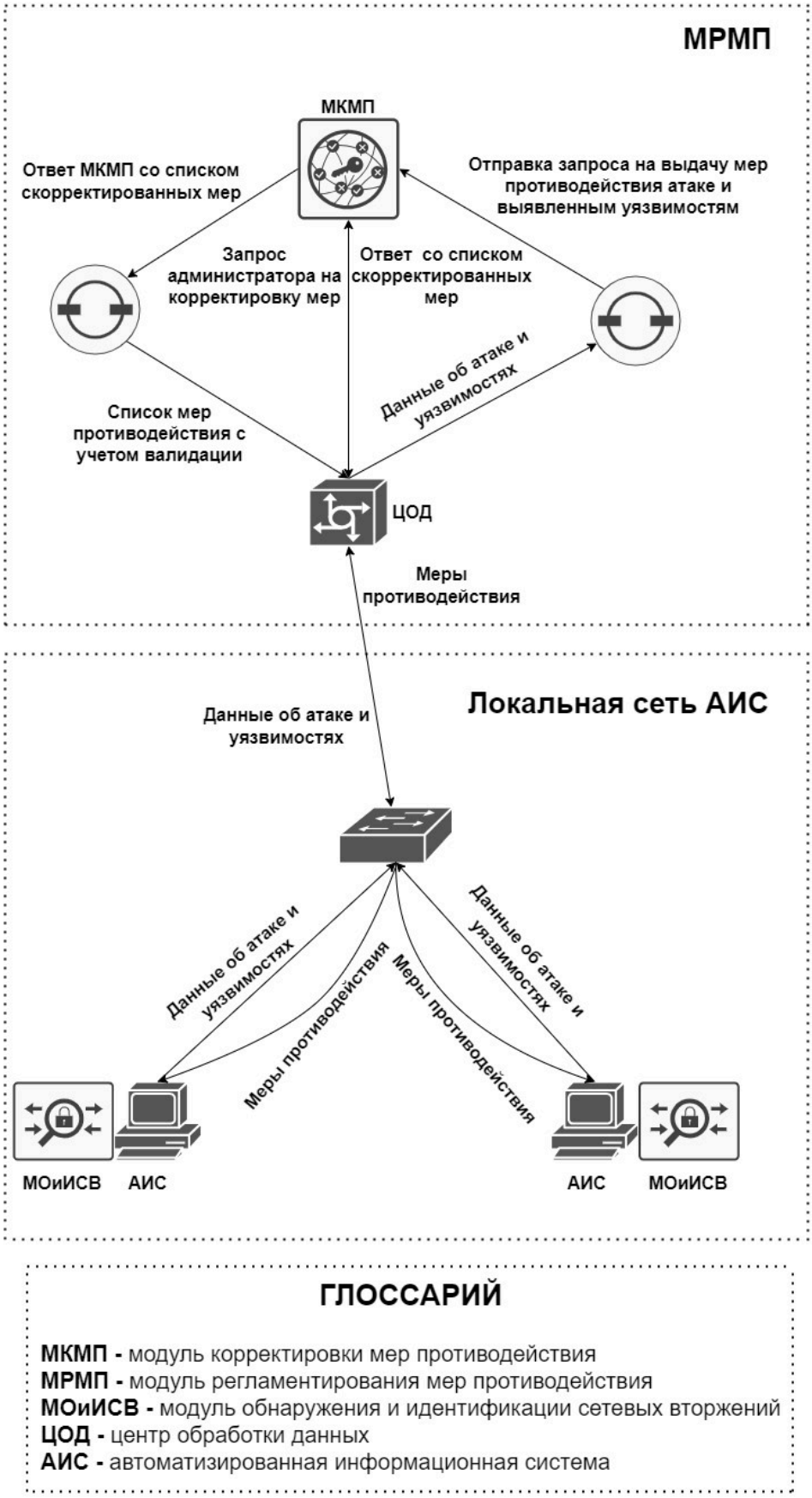

Рис. 2. Архитектура автоматизированной системы обнаружения и идентификация сетевых вторжений с регламентацией мер противодействия

При получении UDP-пакета, содержащего сведения об атаке и уязвимостях, МРМП посредством API производит парсинг данных с размещением их в соответствующий массив, который далее пойдет на вход модуля корректировки мер противодействия (МКМП). В данном модуле сначала администратор получит первичные меры противодействия атаке, реализуемой на хосте, а также рекомендации по смягчению ущербов. После чего будет предоставлена возможность взаимодействия с нейросетевой реализацией для корректировки мер, исходя из требований администратора безопасности.

В результате получения мер, которые удовлетворяют потребностям администратора безопасности, появится возможность применить данные интеллектуальные подсказки в рамках локальной сети и в случае успешного исхода очистить чат хоста, а в случае неудачи продолжить отправку запросов для дальнейшей корректировки.

Взаимодействие с языковой моделью

Исходя из рис. 2, можно сделать вывод о том, что МКМП является базисным элементом МРМП, и от качества его реализации напрямую будет зависеть точность выдаваемых интеллектуальных подсказок. Перед описанием архитектуры данного элемента необходимо отметить, что она выбрана таким образом, чтобы обеспечить изолированность данной системы и независимость от технологий глобальных корпораций. С появлением крупных языковых моделей их создатели начали выпускать API, позволяющие взаимодействовать с их нейросетями и внедрять в свои проекты. Минусом такого варианта является высокая стоимость выдаваемых ответов, так как пользователь платит за токены, которые идут как на вход, так и на выход. В табл. 1 приведен расчет стоимости использования API для языковых моделей Yandex, OpenAI, Mistral, Сбер при генерации 1 миллиона токенов текста, с учетом того, что на вход было подано точно такое же число токенов [2-5].

Таблица 1

Стоимость использования API различных языковых моделей [2-5]

Компания	**Модель**	**Стоимость 1 млн. токенов**
Yandex	YandexGPT Lite	200 руб.
	YandexGPT Pro	7200 руб.
Сбер	GigaChat-Plus	400 руб.
	GigaChat-Pro	1500 руб.
OpenAI	GPT-4	90 $
	GPT-3.5 Turbo	2 $
Mistral	mistral-small	4 $
	mistral-medium	10.8 $
	mistral-large	16 $

Также недостатком данного подхода является невозможность дообучения модели под конкретные нужды в большинстве вариантов. Лишь Яндекс предоставляет возможность прямого дообучения модели. В остальных случаях необходимо использовать технологию Langchain, предусматривающую добавление в запрос данных касательно сферы применения нейросети. При этом очевидно, что количество входных токенов текста будет увеличено по сравнению с обычными запросами, что приведет к повышению стоимости использования API.

Поэтому в рамках данного исследования был выбран другой путь организации взаимодействия с крупной языковой моделью. Существует достаточно ресурсов, таких как HuggingFace [6], магазин LM Studio [7] другие, на которых можно найти открытые языковые модели. К сожалению, развертывание крупной языковой модели требует серьезных ресурсов, так для стабильной работы нейросети с 13 миллиардами параметров необходимо как минимум 24 ГБ оперативной памяти. Однако для большинства известных моделей

существуют оптимизации. Например, для крупной языковой модели Llama 2 70 B существуют версии Llama 2 13 B, Llama 2 7 B и другие. При этом оптимизация представляет собой сокращение материала машинного обучения для нейросети, причем в первую очередь отбрасываются материалы языков отличных от английского. За счет такого подхода происходит выигрыш по ресурсам и небольшие потери по качеству выдаваемых ответов для английского языка, но адекватное общение на других языках становится, по сути, невозможным. Однако существует несколько моделей, имеющих хорошую оптимизацию, при этом обученных на русифицированных данных. Одной из таких моделей является saiga_mistral_7b_lora, обученная российским блогером Ильей Гусевым [8]. В качестве базовой модели была выбрана хорошо оптимизированная mistral_7b, которая по многим всем тестам превосходила модель Llama 2 на 13 миллиарда параметров и по многим Llama 2 на 34 миллиардов параметров. Причем для дообучения данной модели имеется документация, что делает процесс ориентации под конкретную специфику более простым [1].

Структурная схема модуля корректировки мер противодействия

В результате выбора варианта взаимодействия с языковой моделью была сформирована структурная схема взаимодействия элементов модуля корректировки мер противодействия сетевым вторжениям. На рис. 3 показано, что языковая модель разворачивается на отдельном сервере с применением функционала приложения LM Studio, которое позволяет обеспечить удобное взаимодействие с web-приложением, запущенном на компьютере администратора безопасности [7]. По пришествии UDP-пакета и его преобразования в массив, содержащий текстовую информацию об атаках и уязвимостях, посредством API формируется скрытый запрос на языковую модель о необходимости выдачи мер противодействия. Ответ модель направляет на запущенное web-приложение администратора безопасности, и далее администратор имеет возможность общаться с нейросетью в форме чата, сохраняя при этом историю и получая интеллектуальные подсказки на основание ранее полученной информации.

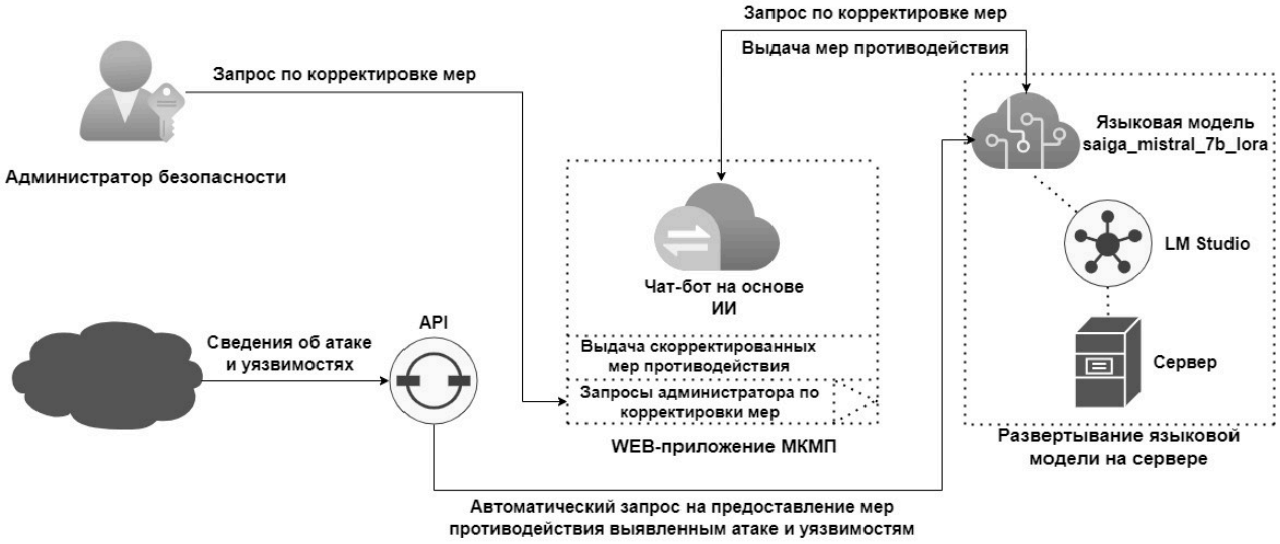

Рис. 3. Структурная схема взаимодействия элементов внутри модуля корректировки мер противодействия

При этом web-приложение необходимо организовать таким образом, чтобы администратор безопасности имел возможность получения интеллектуальных подсказок по каждому из хостов локальной сети по отдельности. Для удовлетворения данной потребности была выделена отдельная вкладка, что показано на рис. 4, появляющаяся автоматически на основании информации об IP-адресах АИС. Причем в каждой вкладке хранится отдельная история чата, что позволяет сформировать меры противодействия с учетом специфики атаки и уязвимостей конкретного хоста.

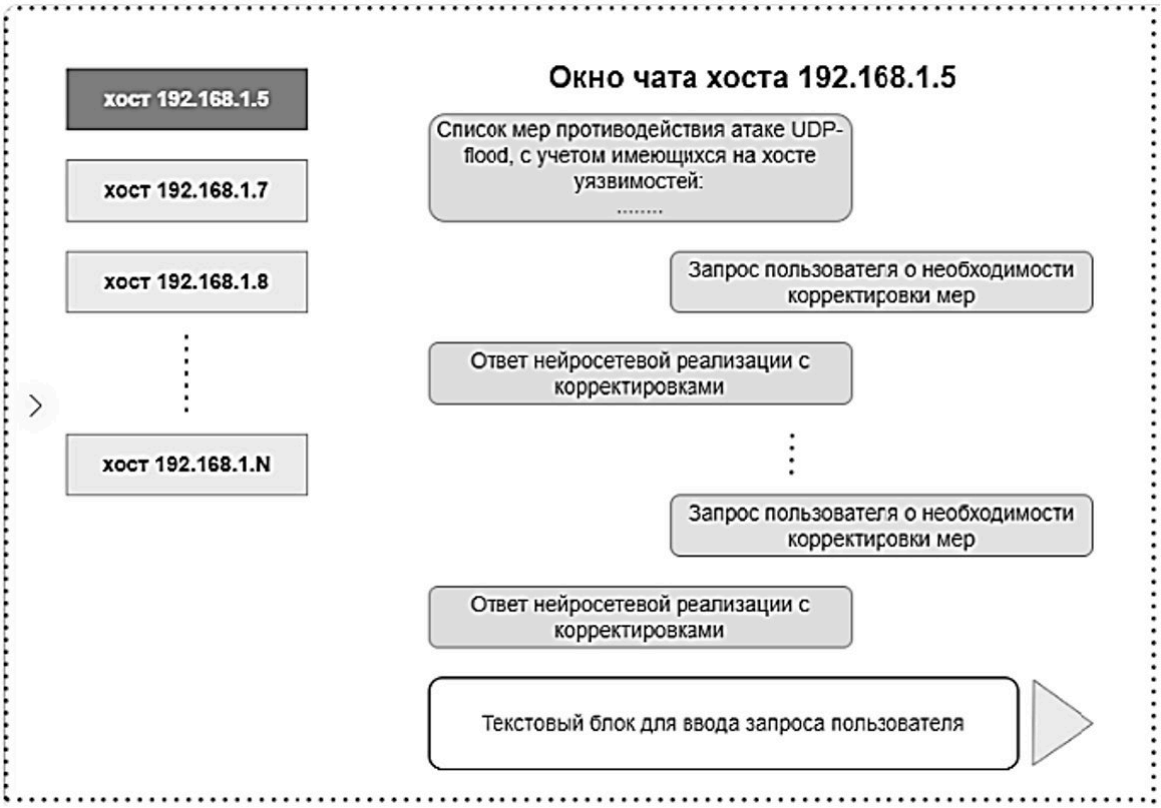

Рис. 4. Структурная схема web-приложения администратора безопасности

Машинное обучение локальной языковой модели

Для корректировки ответов нейросетевого модуля регламентации мер противодействия под специфику сферы ИБ необходимо провести дополнительную настройку языковой модели. Решение этой задачи можно осуществить двумя способами:

– применение концепции машинного обучения моделей fine-tuning;

– использование технологии RAG.

Первый способ представляет собой дополнительное обучение готовой языковой модели на новых данных, в то время как RAG основывается на добавлении к запросам пользователя информации из заранее сформированной базы данных.

Из-за того, что запросы, содержащие большее количество токенов обрабатываются дольше, рациональнее использовать концепцию fine-tuning, при которой скорость выдачи ответов увеличится. Для применения вышеуказанного подхода на практике необходима заранее заготовленная база знаний, приведенная к формату «неправильный json» [9]. Такой формат предусматривает запись каждого элемента с новой строки в фигурных скобках, в которых заключен словарь с тремя ключами «system», «user», «bot», что показано на рис. 5.

```
{ "system":"поведение нейросети", "user": "запрос пользователя", "bot":"ответ нейросети"}
{ "system":"поведение нейросети", "user": "запрос пользователя", "bot":"ответ нейросети"}
{ "system":"поведение нейросети", "user": "запрос пользователя", "bot":"ответ нейросети"}
{ "system":"поведение нейросети", "user": "запрос пользователя", "bot":"ответ нейросети"}
{ "system":"поведение нейросети", "user": "запрос пользователя", "bot":"ответ нейросети"}
{ "system":"поведение нейросети", "user": "запрос пользователя", "bot":"ответ нейросети"}
```

Рис. 5. Формат базы знаний для дополнительного обучения локальной языковой модель

Первый ключ «system» характеризует глобальное поведение нейросети с описанием специфики сферы, в которой она применяется. Оставшиеся ключи необходимы для описания запроса пользователя и ответа нейросетевого бота [8].

Наполнение базы знаний, необходимой для дополнительного обучения языковой модели, стоит проводить на основании сформированной БД о компьютерных атаках и уязвимостях наряду с мерами противодействия по каждой паре. Генерацию этих пар и обновление БД весьма удобно проводить посредством модуля автоматизированного парсинга и аккумуляции данных из баз открытого доступа. Причем в качестве хранилищ, из которых модуль будет вытягивать информацию выступают базы CAPEC и БДУ. БД будет оформлена в виде файла .xlsx, содержащего следующие колонки:

- наименование атаки согласно CAPEC;
- название атаки;
- уязвимость;
- меры реагирования;
- меры ликвидации последствий.

Стоит отметить, что меры реагирования и ликвидации последствий будут записываться для каждой пары атака-уязвимость. Причем запросы пользователя, подаваемые на обучение языковой модели, предполагается делать как по наименованию атаки согласно CAPEC [10], так и по ее названию для более широкого понимания нейросети данной классификации и обеспечения возможности выдачи валидных мер при вводе различных запросов.

Для более быстрого создания базы знаний машинного обучения предполагается использование python-скрипта. Он будет парсить данные из первичной БД, содержащей информацию об атаках, уязвимостях и мерах реагированию, и формировать на их основе словари, наполняющие базу знаний машинного обучения.

Заключение

В статье предлагаются решения в области ИБ, направленные на регламентирование мер противодействия сетевым вторжениям с применением нейросетевых технологий и конкретно крупных языковых моделей в локальном исполнении. Такой вариант является наиболее предпочтительным по сравнению с другими вариантами взаимодействия с крупными языковыми моделями в виду своей автономности, изолированности и конкретизации под специфику области интеллектуальных подсказок для регламентирования мер киберпротиводействия.

Применение подходов и инструментов, выявленных в процессе исследования позволит качественно формулировать меры противодействия, учитывая при этом потребности администратора безопасности. Это в конечном итоге повысит уровень защищенности АИС и может быть весьма полезным при подготовке специалистов по защите информации.

Список литературы

1. Сайга-Мистраль – третья русская нейросеть после YaGPT и GigaChat, публично доступная по API // Хабр : сайт. URL: https://habr.com/ru/articles/767588/ (дата обращения: 13.05.2024).

2. Тарифы и оплата Gigachat // Сбербанк : сайт. URL: https://developers.sber.ru/docs/ru/gigachat/api/tariffs (дата обращения: 13.05.2024).

3. Pricing // OpenAI : сайт. URL: https://openai.com/api/pricing/ (дата обращения: 13.05.2024).

4. Правила тарификации для Yandex Foundation Models // YandexCloud : сайт. – URL: https://yandex.cloud/ru/docs/foundation-models/pricing (дата обращения: 13.05.2024).

5. Models. Getting start // Mistral : сайт. URL: https://docs.mistral.ai/getting-started/models/ (дата обращения: 13.05.2024).

6. Как приручить нейросеть: практический опыт // Хабр : сайт. URL: https://habr.com/ru/companies/reksoft/articles/792496/ (дата обращения: 13.05.2024).

7. IlyaGusev/saiga_mistral_7b_lora // Hugging Face : сайт. URL: https://huggingface.co/IlyaGusev/saiga_mistral_7b_lora (дата обращения: 13.05.2024).

8. Дообучение saiga2_7b_lora // Хабр : сайт. URL: https://habr.com/ru/articles/776872/ (дата обращения: 13.05.2024).

Финансовый университет при Правительстве Российской Федерации
Financial University under the Government of the Russian Federation

Воронежский государственный технический университет
Voronezh State Technical University

Поступила в редакцию 15.05.2024

Информация об авторах

Григорий Александрович Остапенко – д-р техн. наук, профессор, Финансовый университет при Правительстве Российской Федерации, e-mail: ost@fa.ru

Алексей Павлович Васильченко – аспирант, Финансовый университет при Правительстве Российской Федерации, e-mail: rainichek@yandex.ru

Александр Алексеевич Остапенко – аспирант, Воронежский государственный технический университет, e-mail: alexostap123@gmail.com

Александр Александрович Ноздрюхин – студент, Воронежский государственный технический университет, e-mail: sfrvvv@yandex.ru

Данила Сергеевич Покудин – студент, Воронежский государственный технический университет, e-mail: danilapokudin2014@yandex.ru

Никита Николаевич Корвяков – студент, Воронежский государственный технический университет, e-mail: korvyakov48@yandex.ru

NEURAL NETWORK MODULE FOR REGULATING COUNTRMEASURES

G.A. Ostapenko, A.P. Vasilchenko, A.A. Ostapenko, A.A. Nozdriuhin, D.S. Pokudin, N.N. Korvyakov

This article discusses the basics of the functioning of the neural network module for regulating measures to counter network intrusions. The architecture of this module is described, including their interaction with the network intrusion detection and identification module. Instructions for deploying a local version of the neural network are presented and a block diagram of a web application is created that allows the security administrator to receive intelligent tips as part of the regulation of countermeasures.

Keywords: neural network, language model, countermeasures, regulation of countermeasures, security

Submited 15.05.2024

Information about authors

Grigoriy A. Ostapenko – Dr. Sc. (Technical), Professor, Financial University under the Government of the Russian Federation, e-mail: ost@fa.ru

Alexsey P. Vasilchenko – graduate student, Financial University under the Government of the Russian Federation, e-mail: rainichek@yandex.ru

Aleksandr A. Ostapenko – graduate student, Voronezh State Technical University, e-mail: alexostap123@gmail.com

Aleksandr A. Nozdriuhin – student, Voronezh State Technical University, e-mail: sfrvvv@yandex.ru

Danila S. Pokudin – student, Voronezh State Technical University, e-mail: danilapokudin2014@yandex.ru

Nikita N. Korvyakov – student, Voronezh State Technical University, e-mail: korvyakov48@yandex.ru