ОЦЕНКА ЗАЩИЩЁННОСТИ ACTIVE DIRECTORY С ИСПОЛЬЗОВАНИЕМ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Булгакова Е. В.[1], Богданов Е. А.[2], Кубанков А. Н.[3]

DOI: 10.21681/2311-3456-2025-6-58-68

Цель исследования: *применение методов искусственного интеллекта, в частности машинное обучение и нейронные сети, для оценки защищённости системы Microsoft Active Directory и выявления факторов, влияющих на уровень безопасности корпоративных сетей. Работа направлена на разработку и сравнение алгоритмов, способных прогнозировать степень уязвимости пользователей и подсистем Active Directory.*

Методы исследования: *исследование основывается на методах машинного обучения, включающие метрические алгоритмы (линейная регрессия, метод ближайших соседей, дерево решений, случайный лес, градиентный бустинг) и нейронные сети, реализованные в среде Jupyter Notebook с применением библиотек pandas, sklearn и keras. На основе подготовленного датасета выполнена стандартизация и нормализация параметров, отражающих конфигурацию пользователей Active Directory. Для проверки эффективности алгоритмов проведено сравнение по критериям точности прогнозирования и среднеквадратичной ошибки.*

Результаты исследования: *проведён анализ факторов, влияющих на защищённость корпоративной инфраструктуры Active Directory, включая тип операционной системы, длину и срок действия пароля, уровень привилегий, параметры делегирования и наличие предварительной аутентификации Kerberos. На основе подготовленного датасета были реализованы и протестированы различные алгоритмы машинного обучения. Результаты показали, что модель дерева решений продемонстрировала наилучшие показатели — точность прогнозирования 0,96 и среднеквадратичную ошибку 0,091, что свидетельствует о её высокой эффективности в задаче оценки защищённости Active Directory. В дополнение была разработана модель нейронной сети и подтверждена её способность корректно обрабатывать параметры Active Directory и определять степень безопасности пользователей этой системы. Полученные результаты указывают на перспективность применения технологий искусственного интеллекта для автоматизации анализа уязвимостей и прогнозирования рисков информационной безопасности в корпоративных сетях.*

Научная новизна: *научная новизна исследования заключается в разработке и апробации интегрированного подхода к оценке защищённости Active Directory на основе машинного обучения и нейронных сетей. Предложено использование интеллектуальных моделей для прогнозирования уровня безопасности пользователей с учётом комплексных параметров инфраструктуры Active Directory, что позволяет формировать автоматизированную систему раннего предупреждения об уязвимостях корпоративных сетей.*

Ключевые слова: *Active Directory, информационная безопасность, искусственный интеллект, машинное обучение, нейронные сети, дерево решений, анализ защищённости, уязвимости, корпоративные сети*

Введение

В настоящее время организации все чаще внедряют гибридные форматы работы. Важнейшей компонентой таких инфраструктур является служба каталогов Active Directory (AD), отвечающая за централизованное администрирование учетных записей пользователей, ресурсов и групп.

Увеличение сложности корпоративных ИТ-систем влечет за собой повышение требований к их безопасности. Риск несанкционированного доступа, применение устаревшего программного обеспечения создают условия для компрометации Active Directory.

В условиях постоянного роста объемов данных, традиционные методы анализа и аудита теряют в результативности. Это связано с человеческим фактором и с ограниченной масштабируемостью существующих инструментов. Использование технологий искусственного интеллекта открывает новые перспективы для автоматизации оценки состояния безопасности инфраструктуры Active Directory, давая возможность выявлять отклонения и векторы атак злоумышленников.

В статье будут рассмотрены подходы к оценке защищённости Active Directory с использованием методов искусственного интеллекта.

Параметры рассматриваемой базы данных

Для анализа системы были определены основные параметры, которые влияют на безопасность корпоративной сети, и присвоены им значения (вес), которые будет влиять на конечное значение безопасности, целевым признаком будет относительная

1 Булгакова Елена Валерьевна, кандидат юридических наук, доцент, заместитель заведующего кафедрой информационной безопасности по научной работе, Финансовый университет при Правительстве РФ, г. Москва, Россия. E-mail: koordinator-proekta@mail.ru

2 Богданов Евгений Александрович, Ph.D., заведующий кафедрой информационной безопасности, Финансовый университет при Правительстве РФ, г. Москва, Россия. E-mail: eabogdanov@fa.ru

3 Кубанков Александр Николаевич, доктор военных наук, кандидат технических наук, профессор, профессор кафедры информационной безопасности, Финансовый университет при Правительстве РФ, г. Москва, Россия. E-mail: kan9991@gmail.com

защищенность системы. Более высокие значения веса – больше шанс компрометации системы. Таблица с весами для каждого параметра (табл. 1):

Таблица 1.
Параметры датасета с весами каждого значения

Параметр	Значение	Вес
Operating_ System	Windows XP	30
	Windows 7	10
	Windows 10	5
	Windows 11	0
	Ubuntu	0
Relationship	Admins	10
	Peers	5
	Exec	5
	Users	5
	Active	3
Expiring_pass	True	0
	False	3
Pass_Length	Short	5
	Medium	3
	Long	0
Change_pass	True	0
	False	3
Auth_Kerberos	True	0
	False	10
Updating	True	0
	False	3
Privilege	High	50
	Medium	30
	Low	20
Delegation_ Param	Const	0
	Resource	3
	Unconst	5
Safety	0	–
	1	

Описание параметров:

- Operating System отвечает за выбранную операционную систему.
- Relationship – параметр, показывающий, кем является пользователь:
  - Admins – администраторы систем Active Directory;
  - Peers – пользователи, близкие по полномочиям и по местоположению в сети к администраторам;
  - Exec – сотрудники, имеющие ограниченный доступ к правам администратора;
  - Users – обычные пользователи.
  - Active – сотрудники, работающие за устройством на данный момент.
- Expiring_pass – параметр, показывающий просрочен пароль или нет.
- Pass_Length – длина пароля; параметр важен, так как пароль может быть получен методом полного перебора [1, 2].
- Change_pass – менял ли пользователь пароль.
- Auth_Kerberos – требует ли учетная запись предварительную аутентификацию Kerberos [3].
- Updating – показатель, отвечающий за обновление операционной системы.
- Privilege – доступность к файлам во всей структуре.
- Delegation_Param – делегирование функций.
- Safety – оценка защищённости системы AD (целевой параметр). 0 – система защищена, иначе – 1.

Подготовка данных в датасете

Для подготовки базы данных, с которой мы будем взаимодействовать, воспользуемся программным обеспечением Jupyter Notebook [4]. Для корректной работы с датасетом загружаем необходимые библиотеки, в первую очередь библиотеку Pandas [5].

Далее загружаем датасет и находим средние значения для каждого параметра (рис. 1).

```
#Средние значения для каждого параметра
df.mean()
```

```
Operating_System    6.955
Expiring_pass       1.473
Change_pass         1.374
Auth_Kerberos       4.740
Pass_Length         2.722
Updating            1.926
Privilege          30.520
Delegation_Param    2.585
Relantionship       4.574
Safety              0.557
dtype: float64
```

Рис. 1. Средние значения для каждого параметра

Стандартизируем и нормализуем сданные для удобной работы, здесь понадобятся соответственно StandardScaler и MinMaxScaler [6] (рис. 2, 3).

Представим наши данные в виде графиков (рис. 4).

DOI: 10.21681/2311-3456-2025-6-58-68

```
scaler = StandardScaler()
pd.set_option('display.float_format',lambda x: '%0.3f' % x)
std_df = scaler.fit_transform(df)
origin_df = scaler.inverse_transform(std_df)
```

```
std_df
```

```
array([[ 2.61347413, -0.98215912,  1.08784471, ...,  1.18223728,
         0.28652464,  0.89181396],
       [ 0.34532561,  1.01816496, -0.91924885, ..., -1.26545896,
         0.28652464, -1.1213101 ],
       [ 2.61347413, -0.98215912, -0.91924885, ...,  1.18223728,
         0.28652464,  0.89181396],
       ...,
       [ 2.61347413, -0.98215912,  1.08784471, ..., -1.26545896,
         0.28652464,  0.89181396],
       [-0.22171152, -0.98215912, -0.91924885, ..., -1.26545896,
         0.28652464, -1.1213101 ],
       [-0.78874865, -0.98215912, -0.91924885, ...,  0.20315879,
         0.28652464, -1.1213101 ]])
```

Рис. 2. Стандартизация данных в датасете

```
mm = MinMaxScaler()
mm_df = mm.fit_transform(df)
origin_df = mm.inverse_transform(mm_df)
```

```
mm_df
```

```
array([[1.        , 0.        , 1.        , ..., 1.        , 0.28571429,
        1.        ],
       [0.33333333, 1.        , 0.        , ..., 0.        , 0.28571429,
        0.        ],
       [1.        , 0.        , 0.        , ..., 1.        , 0.28571429,
        1.        ],
       ...,
       [1.        , 0.        , 1.        , ..., 0.        , 0.28571429,
        1.        ],
       [0.16666667, 0.        , 0.        , ..., 0.        , 0.28571429,
        0.        ],
       [0.        , 0.        , 0.        , ..., 0.6       , 0.28571429,
        0.        ]])
```

Рис. 3. Нормализация данных в датасете

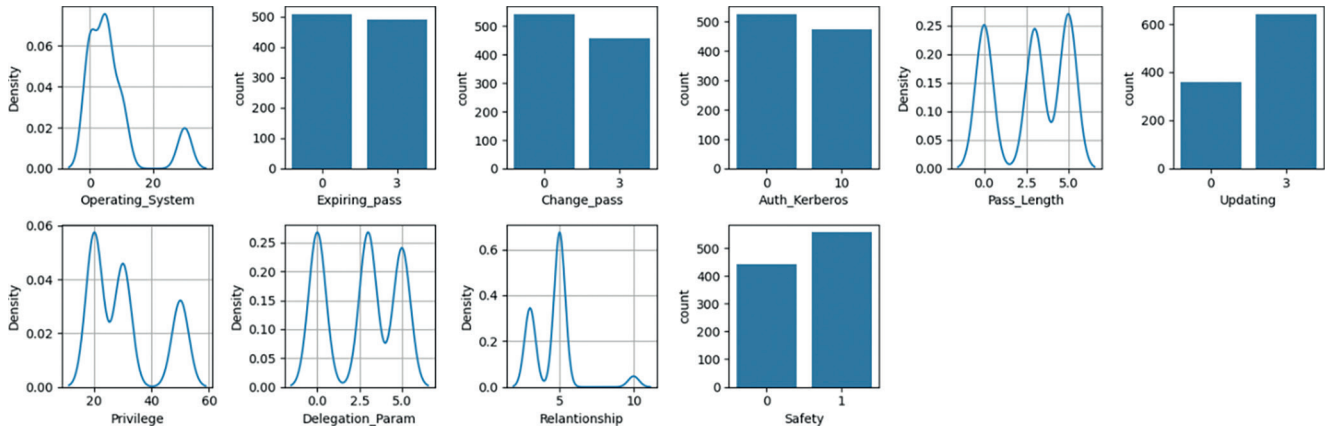

Рис. 4. Графики исходных данных датасета

Реализация метрических алгоритмов

Для реализации метрических алгоритмов нам понадобится библиотека sklearn [7] (рис. 5).

```
lr = LinearRegression() #Линейная регрессия
knr = KNeighborsRegressor() #Метод ближайших соседей
dt = DecisionTreeRegressor() #Дерево решений
rf = RandomForestRegressor() #Случайный лес
gbr = GradientBoostingRegressor() #Градиентный бустинг
```

Рис. 5. Ввод метрических алгоритмов в систему

Разделим датасет на две отдельных части, в первой будут столбцы, содержащие параметры для оценивания Active Directory, а во второй находятся данные, которые отображают защищенность системы на основе значений первой [8] (рис. 6).

```
y = df['Safety']
x = df.drop('Safety', axis =
```

Рис. 6. Разделение датасета на две части

Для корректной работы алгоритмов машинного обучения разделяем значения на тестовую и обучающую выборки [9] (рис. 7).

```
#Создание обучающей и тестовой выборки
from sklearn.model_selection import train_test_split
xtrain,xtest,ytrain,ytest = train_test_split(x,y,test_size = 0.30, random_state = 42)
```

Рис. 7. Разделение данных на обучающую и тестовую выборки

Выведем графики, обозначим среднюю квадратическую ошибку и эффективность для каждой модели, воспользовавшись библиотеками matplotlib [10] и sklearn.metrics (рис. 8–17). Графики представляют собой соотношение плотности функций от параметра защищённости (табл. 1).

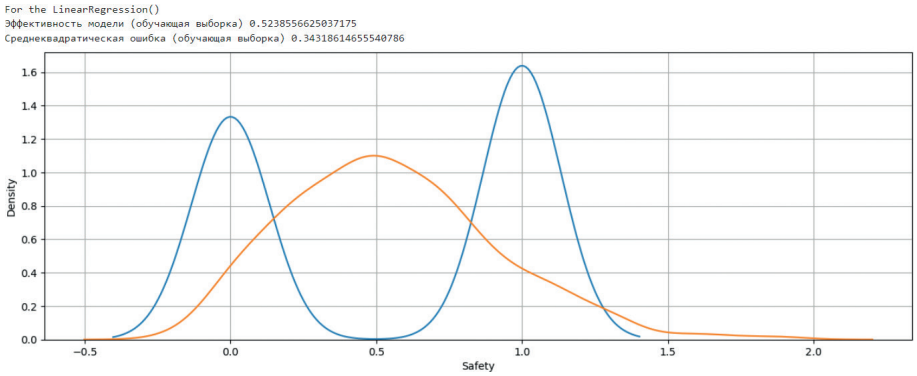

Рис. 8. График линейной регрессии для обучающей выборки

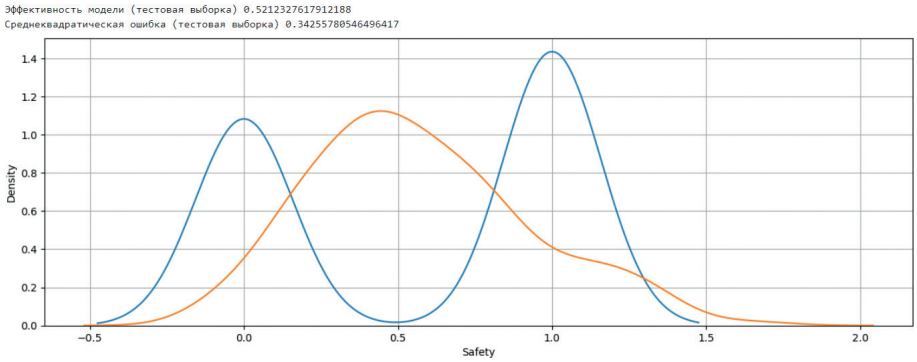

Рис. 9. График линейной регрессии для тестовой выборки

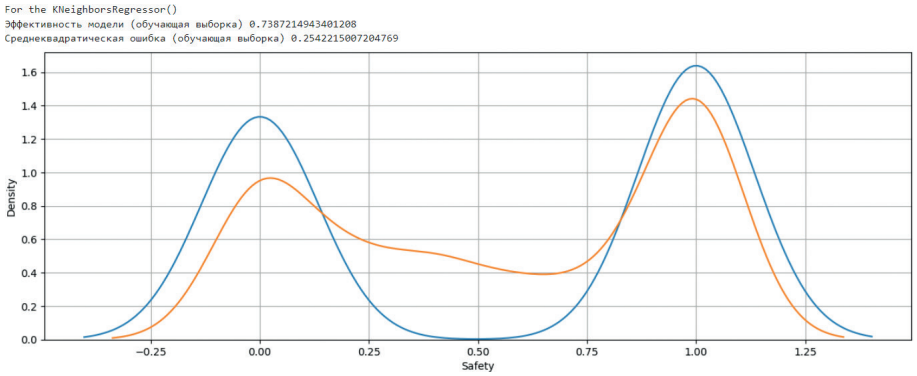

Рис. 10. График метода ближайших соседей для обучающей выборки

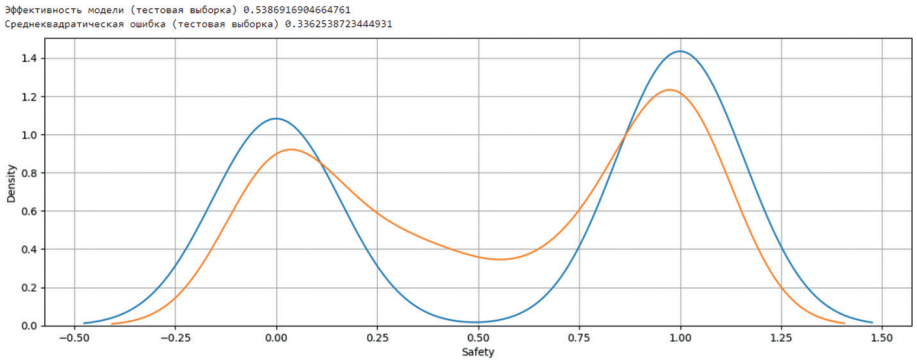

Рис. 11. График метода ближайших соседей для тестовой выборки

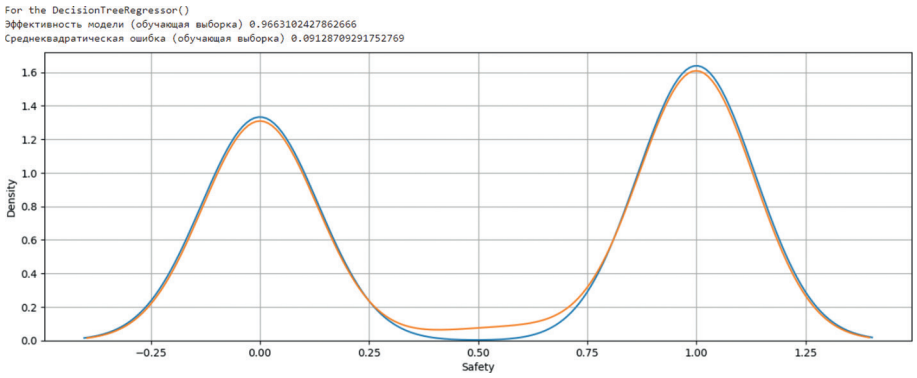

Рис. 12. График дерева решений для обучающей выборки

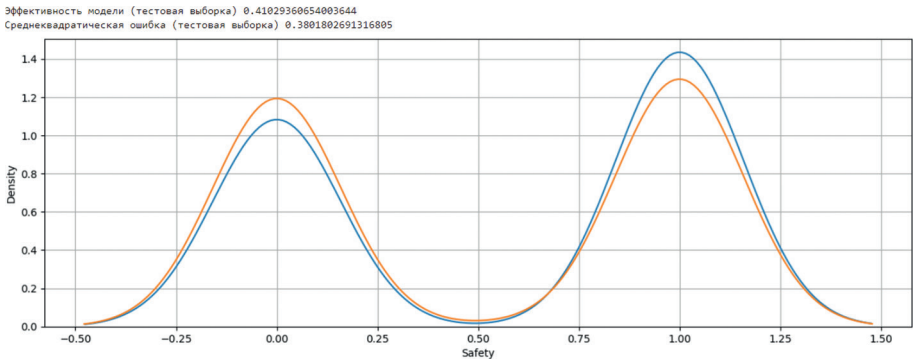

Рис. 13. График дерева решений для тестовой выборки

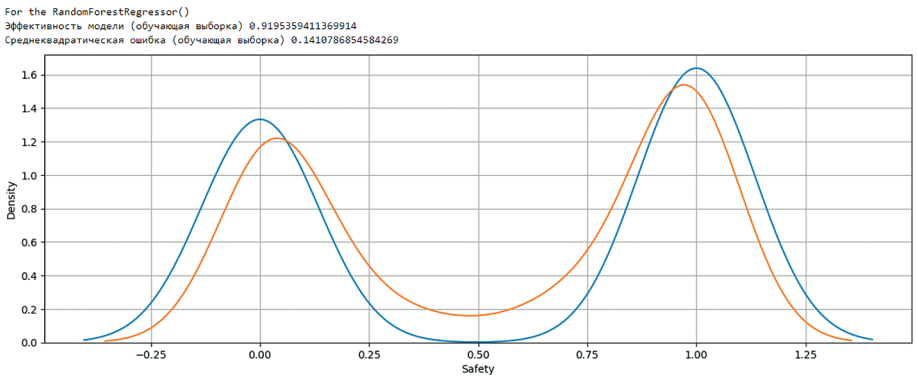

Рис. 14. График случайного леса для обучающей выборки

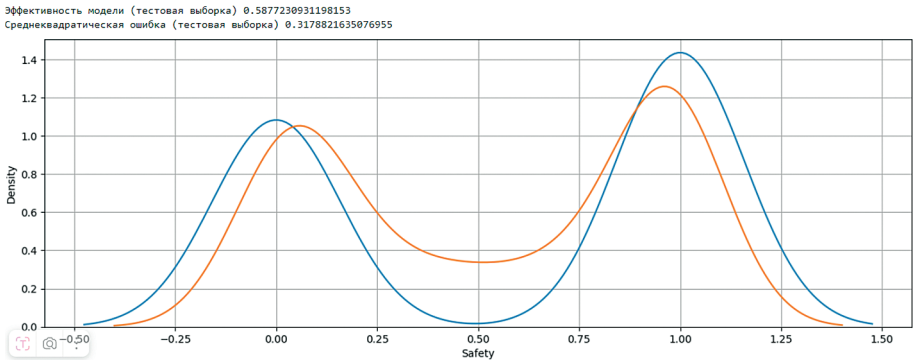

Рис. 15. График случайного леса для тестовой выборки

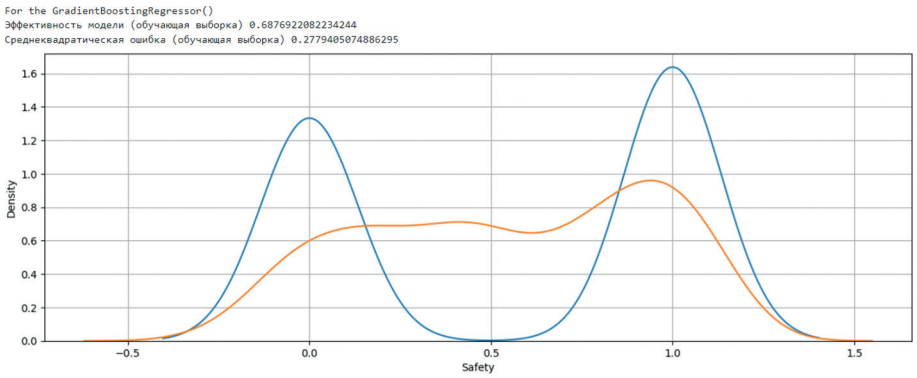

Рис. 16. График градиентного бустинга для обучающей выборки

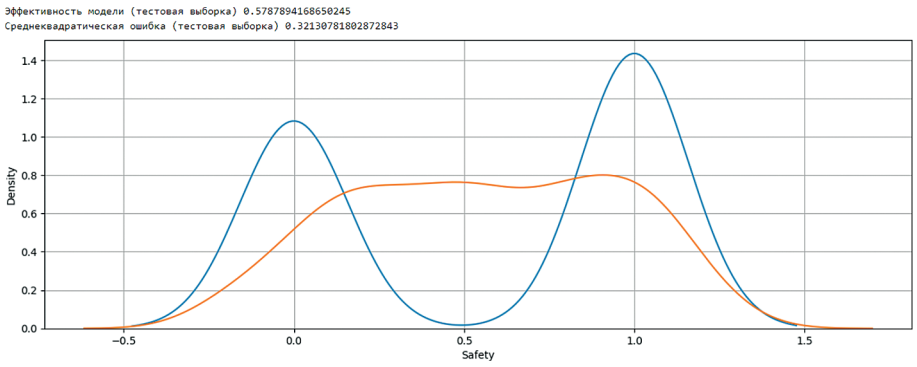

Рис. 17. График градиентного бустинга для тестовой выборки

Как видно из графиков, среди представленных методов МО дерево решений наиболее полно описывает зависимость целевого параметра Safety (защищённость системы AD) от входных параметров. Обратная ситуация возникает с линейной регрессией и градиентным бустингом, которые эту зависимость объясняют плохо.

Реализация работы нейронной сети

Для полноты исследования реализуем модель нейронной сети, которая на основе введенных данных сможет вычислить, защищен ли пользователь от различных атак злоумышленников, соответственно оценивая защищенность Active Directory.

Проведем те же действия над данными, как и с метрическими алгоритмами, разделим их на две части, сначала на входные и выходные параметры, затем на обучающую и тестовую выборки.

Сначала используем Keras (рис. 18):

```
import tensorflow
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
```

Рис. 18. Импорт Keras

Для добавления слоев в первую очередь инициализируем нейронную сеть в системе. Добавим объект нейронной сети типа sequential (рис. 19).

DOI: 10.21681/2311-3456-2025-6-58-68

```
classifier = Sequential()
```

Рис. 19. Инициализация нейронной сети

```
classifier.add(Dense(units = 6, activation = 'relu', input_dim = 9, kernel_initializer = 'uniform'))
```

Рис. 20. Интеграция входного и первого скрытого слоев

```
classifier.add(Dense(units = 6, activation = 'relu', kernel_initializer = 'uniform'))
```

Рис. 21. Интеграция второго скрытого слоя

Добавим входной слой и первый скрытый слой, использующиеся при обработке данных (рис. 20). Слой создается с нелинейной функцией активации «relu», kernel_initializer = 'uniform' является параметром по умолчанию и создает вес слоя [11, 12].

После чего добавляем второй скрытый слой, установив такие же значения, как и при создании первого слоя (рис. 21).

Заключительным слоем будет выходной, в нем будет присутствовать один нейрон [13].

Проводим тренировку нейронной сети, которая будет реализована на основе тренировочной выборки (рис. 22).

```
classifier.fit(X_trainNN, y_trainNN, batch_size = 10, epochs = 100)
```

```
Epoch 4/100
70/70 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - accuracy: 0.8358 - loss: 0.5424
Epoch 5/100
70/70 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - accuracy: 0.8242 - loss: 0.5025
Epoch 6/100
70/70 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - accuracy: 0.8757 - loss: 0.4382
Epoch 7/100
70/70 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - accuracy: 0.8632 - loss: 0.4206
Epoch 8/100
70/70 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.8950 - loss: 0.3717
Epoch 9/100
70/70 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - accuracy: 0.9087 - loss: 0.3473
Epoch 10/100
70/70 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - accuracy: 0.8807 - loss: 0.3482
Epoch 11/100
70/70 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - accuracy: 0.9117 - loss: 0.3108
Epoch 12/100
70/70 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - accuracy: 0.8981 - loss: 0.3206
```

Рис. 22. Тренировка нейронной сети

После тренировки нейронной сети получаем систему, позволяющую определять защищенность того или иного пользователя в корпоративной сети.

Нейронная сеть была сформирована из входного, выходного и двух скрытых слоев в каждом слое было по 6 нейронов, всего использовалось 100 эпох. Тестовая выборка составляла 0.3 от всех исходных данных, что является наиболее оптимальным. При наличии большего количества, это значение будет варьироваться.

Тестирование нейронной сети

После обучения системы выведем прогноз для тестовой выборки, в которой содержатся получившиеся значения и выводы для каждого набора параметров (рис. 23).

```
y_predNN = classifier.predict(X_testNN)
y_predNN
```

```
10/10 ━━━━━━━━━━━━━━━━━━━━ 0s 14ms/step

array([[0.97422767],
       [0.63163245],
       [0.80098426],
       [0.03812398],
       [0.13650697],
       [0.6287609 ],
       [0.99998623],
       [0.01737837],
       [0.01010334],
       [0.62079155],
       [0.15893725],
       [0.99894065],
       [0.9999503 ],
       [0.11676516],
       [0.982782  ],
       [0.05276635],
```

Рис. 23. Прогноз для тестовой выборки

Получившиеся значения представим в виде булевых значений (защищенная система или нет). Для разделения используется значение 0,5 и выводы больше него означают, что система не защищена (рис. 24).

Вводим матрицу несоответствий [14, 15]. Для реализации используется метод confusion_matrix, который подсчитывает все данные, а также тестовая выборка (рис. 25).

Получившиеся значения в большинстве отображаются как правильные (ошибок I и II рода относительно немного), это означает, что алгоритм работает корректно.

Проведем анализ для отдельно взятых сгенерированных пользователей для того, чтобы посмотреть правильность работы алгоритма [16] (рис. 26).

```
y_predNN = (y_predNN > 0.5)
y_predNN
```

```
array([[ True],
       [ True],
       [ True],
       [False],
       [False],
       [ True],
       [ True],
       [False],
       [False],
       [ True],
       [False],
       [ True],
       [ True],
       [False],
       [ True],
       [False],
       [False],
       [ True]
```

Рис. 24. Прогноз в виде булевых значений

```
cm = confusion_matrix(y_testNN, y_predNN)
cm
```

```
array([[107,  23],
       [ 28, 142]], dtype=int64)
```

Рис. 25. Формирование матрицы несоответствий

Первый пользователь работает на ОС Windows 7, у него просроченный короткий пароль, который не изменялся в течении 100 дней, система не обновлялась с первичной установки, ограниченный доступ к функциям администратора, ресурсное делегирование, имеет активную сессию (рис. 27).

Второй пользователь работает на ОС Windows 11, система не обновлялась, короткий пароль, нет доступа к функциям администратора, ресурсное делегирование, имеет активную сессию (рис. 28).

Возможные параметры, которые могут быть добавлены в систему

Добавление новых параметров является неотъемлемой частью заполнения датасета, так как в статье представлены лишь некоторые параметры, которые имеют наибольший вес во всем объеме настроек. Злоумышленники обладают огромным набором утилит для влияния и проникновения в систему, увеличив количество входных данных появляется возможность минимизировать шанс ошибки.

Большинство параметров имеет важное значение для безопасности пользователя, более детальный анализ показывает, что имеет смысл добавить список новых настроек, которые помогут в решении основной задачи. Все это может влиять как на пользователя, так и на сервер, к которому он относится. Разберем детальнее отдельно возможные варианты параметров.

Accountdisable – определяет активность учетной записи, является ли данный пользователь отключенным или нет, есть ли у него возможность управлять аккаунтом. В компании может содержаться огромное количество УЗ без работающего владельца. Такие записи в отдельных предприятиях не проверяются и остаются на многие годы без проверок.

Trusted_To_Auth_For_Delegation – параметр осведомляющий о том, что учетная запись делегируется, в данном случае служба использует права пользователя и выполняет проверку от его лица в других удаленных местах сети (рис. 29).

Также для администраторов следует ввести дополнительный параметр, который будет отвечать за доступность участников к сети, подключаться и видеть контроллер домена. В папке C:\Windows\NTDS\ находится файл ntds.dit, который хранит информацию

```
new_prediction1 = classifier.predict(scaler.transform(np.array([[10,3,0,0,5,3,30,3,3]])))
new_prediction2 = classifier.predict(scaler.transform(np.array([[0,0,0,0,5,3,20,3,3]])))
```

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 98ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 70ms/step
```

Рис. 26. Код ввода случайных значений для пользователей

```
new_prediction1
```

```
array([[0.8170041]], dtype=float32)
```

Рис. 27. Вывод решения о первом пользователе

```
new_prediction2
```

```
array([[0.02366159]], dtype=float32)
```

Рис. 28. Вывод решения о втором пользователе

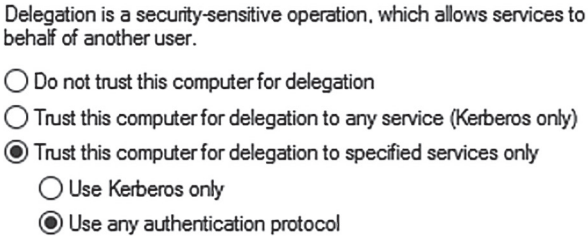

Рис. 29. Включенный параметр Trusted_To_Auth_For_Delegation

о всей локальной сети Active Directory. Чем больше пользователей подключены к котроллеру, тем больше шанс пропустить злоумышленника, который скомпрометирует данные.

Заключение

В условиях роста числа удалённых пользователей традиционные методы администрирования Active Directory, уязвимости, связанные с использованием устаревших операционных систем, слабых паролей, создают серьёзные риски компрометации и требуют применения более интеллектуальных методов анализа.

В результате исследования был сформирован датасет, включающий параметры, влияющие на защищённость пользователей Active Directory, и реализованы различные методы машинного обучения. Сравнительный анализ показал, что с задачей оценки защищённости лучше всего справился алгоритм дерева решений, обеспечив высокую точность прогнозов и низкий уровень ошибок, при этом хуже всего справились алгоритмы линейной регрессии и градиентного бустинга.

Также внимание уделено разработке и обучению нейронной сети. Было показано, как современные инструменты машинного обучения и анализа данных позволяют относительно просто проектировать модели (в частности, нейронных сетей), которые способны обрабатывать большие объёмы данных и выполнять предсказания с высокой точностью.

Результаты работы показывают перспективность применения технологий ИИ для анализа информационных систем и прогнозирования рисков информационной безопасности в режиме реального времени. Так, внедрение методов машинного обучения в процесс оценки защищённости Active Directory позволяет существенно повысить точность и скорость анализа, минимизировать влияние человеческого фактора и обеспечить более высокий уровень устойчивости корпоративной инфраструктуры к современным киберугрозам.

Литература

1. Review of prevention schemes for man-in-the-middle (MITM) attack in vehicular ad hoc networks. Al-Shareeda M. A. International Journal of Engineering and Management Research. 2020. – 6 p. – DOI 10.31033/ijemr.10.3.23.
2. Правоторова А. Ю. Оценка стойкости паролей: сравнительное исследование эффективности длины и сложности / А. Ю. Правоторова // Наука. Инновации. будущее – 2025 : Сборник статей II Международной научно-практической конференции, Петрозаводск, 15 мая 2025 года. – Петрозаводск: Международный центр научного партнерства «Новая Наука» (ИП Ивановская И.И.), 2025. – С. 125–132. – EDN MEPXNB.
3. Khalil Nabab Pinjari. LSTM-Enabled Big Data Security Framework Integrating Kerberos Authentication on AWS for Robust Cloud Protection / Khalil Nabab Pinjari, Abu Zar Muhammad, Yogesh Kumar Sharma // Nanotechnology Perceptions. – 2024. – Vol. 20, No. 7. – DOI 10.62441/nano-ntp.v20i7.4414. – EDN ZHQLIB.
4. Абрамова Е. В. Возможности Google Colab и Jupyter Notebook для решения задач искусственного интеллекта / Е. В. Абрамова, Л. А. Максименко // Регулирование земельно-имущественных отношений в России: правовое и геопространственное обеспечение, оценка недвижимости, экология, технологические решения. – 2023. – № 1. – С. 23–29. – DOI 10.33764/2687-041X-2023-1-23-29. – EDN CYLSKT.
5. Ильичев В. Ю. Анализ массивов данных с использованием библиотеки Pandas для Python / В. Ю. Ильичев, Е. А. Юрик // Научное обозрение. Технические науки. – 2020. – № 4. – С. 41–45. – EDN BKGJHM.
6. Судариков Г. В. Использование библиотеки Pandas для анализа данных / Г. В. Судариков, И. А. Ашмаров // Мир образования - образование в мире. – 2023. – № 1(89). – С. 184–188. – DOI 10.51944/20738536_2023_1_184. – EDN NWHNRO.
7. Florescu D. A Machine Learning Based Software Pipeline to Pick the Variable Ordering for Algorithms with Polynomial Inputs / D. Florescu, M. England // Lecture Notes in Computer Science. – 2020. – Vol. 12097 LNCS. – P. 302–311. – DOI 10.1007/978-3-030-52200-1_30. – EDN JDDNYT.
8. Retnoningsih, E. Mengenal Machine Learning Dengan Teknik Supervised Dan Unsupervised Learning Menggunakan Python / E. Retnoningsih, R. Pramudita // Bina Insani ICT Journal. – 2020. – Vol. 7, No. 2. – P. 156. – DOI 10.51211/biict.v7i2.1422. – EDN TNZXLG.
9. Булгакова Е. В. Проблема точности и объяснимости при внедрении искусственного интеллекта в системы управления информацией и событиями безопасности / Е. В. Булгакова, Д. С. Дойников, А. Н. Кубанков // Наукоемкие технологии в космических исследованиях Земли. – 2025. – Т. 17, № 3. – С. 35–41. – DOI 10.36724/2409-5419-2025-17-3-35-41. – EDN QBTNNT.
10. Магеррамов И. М. Нечеткие подходы к решению задач классификации / И. М. Магеррамов, Т. С. Александрова // Интеллектуальные ресурсы - региональному развитию. – 2021. – № 2. – С. 76–82. – EDN XRMACX.
11. Feng, G. Common Python Data Analysis Method Based on Deep Learning / G. Feng // Journal of Physics: Conference Series. – 2021. – Vol. 2037, No. 1. – P. 012132. – DOI 10.1088/1742-6596/2037/1/012132. – EDN YQAYRU.
12. Abdulkadirov, R. Survey of Optimization Algorithms in Modern Neural Networks / R. Abdulkadirov, P. Lyakhov, N. Nagornov // Mathematics. – 2023. – Vol. 11, No. 11. – P. 2466. – DOI 10.3390/math11112466. – EDN BAUENY.
13. Tao, Ch. Applications of Bayesian Neural Networks in Outlier Detection / Ch. Tao // Big Data. – 2023. – Vol. 11, No. 5. – P. 369–386. – DOI 10.1089/big.2021.0343. – EDN DXZRFS.

14. Iiduka, H. Training deep neural networks using conjugate gradient-like methods / H. Iiduka, Y. Kobayashi // Electronics. – 2020. – Vol. 9, No. 11. – P. 1–25. – DOI 10.3390/electronics9111809. – EDN TMEEXV.
15. Трунов Е. Е. Обнаружение угроз безопасности информации с использованием глубоких нейронных сетей в компьютерных сетях в режиме реального времени / Е. Е. Трунов, С. Г. Клюев // Моделирование, оптимизация и информационные технологии. – 2022. – Т. 10, № 3(38). – С. 12-13. – DOI 10.26102/2310-6018/2022.38.3.011. – EDN MNLGVN.
16. Nakhushev, R. S. Application of the neural networks for cryptographic information security / R. S. Nakhushev, N. V. Sukhanova // Proceedings of the 2020 IEEE International Conference «Quality Management, Transport and Information Security, Information Technologies», IT and QM and IS 2020, Yaroslavl, 07–11 сентября 2020 года. – Yaroslavl, 2020. – P. 421–423. – DOI 10.1109/ITQMIS51053.2020.9322981. – EDN FALTNR.

ASSESSMENT OF ACTIVE DIRECTORY SECURITY USING ARTIFICIAL INTELLIGENCE

Bulgakova E. V.[4], Bogdanov E. A.[5], Kubankov A. N.[6]

Keywords: *Active Directory, information security, artificial intelligence, machine learning, neural networks, decision tree, security assessment, vulnerabilities, corporate networks.*

Purpose of the study: *the study aims to apply artificial intelligence methods, including machine learning and neural networks, to assess the security of Microsoft Active Directory and identify factors affecting the security level of corporate networks. The work focuses on developing and comparing algorithms capable of predicting the vulnerability levels of Active Directory users and subsystems.*

Methods of research: *the study is based on machine learning methods, including metric algorithms (linear regression, nearest neighbour method, decision tree, random forest, gradient boosting) and neural networks, implemented in the Jupyter Notebook environment using the pandas, sklearn and keras libraries. Based on the prepared dataset, the parameters reflecting the configuration of Active Directory users were standardised and normalised. To verify the effectiveness of the algorithms, a comparison was made based on the criteria of prediction accuracy and root mean square error.*

Results: *an analysis was conducted of factors affecting the security of the Active Directory corporate infrastructure, including operating system type, password length and validity period, privilege level, delegation settings, and the presence of Kerberos pre-authentication. Various machine learning algorithms were implemented and tested based on the prepared dataset. The results showed that the decision tree model demonstrated the best performance, with a prediction accuracy of 0.96 and a root mean square error of 0.091, indicating its high effectiveness in assessing Active Directory security. In addition, a neural network model was developed and its ability to correctly process Active Directory parameters and determine the security level of users of this system was confirmed. The results obtained indicate the promise of using artificial intelligence technologies to automate vulnerability analysis and predict information security risks in corporate networks.*

Scientific novelty: *the scientific novelty of the research lies in the development and testing of an integrated approach to assessing Active Directory security based on machine learning and neural networks. The use of intelligent models is proposed to predict the level of user security, taking into account the complex parameters of the Active Directory infrastructure, which allows the formation of an automated early warning system for corporate network vulnerabilities.*

References

1. Review of prevention schemes for man-in-the-middle (MITM) attack in vehicular ad hoc networks. Al-Shareeda M. A. International Journal of Engineering and Management Research. 2020. – 6 p. – DOI 10.31033/ijemr.10.3.23.
2. Pravotorova A. Ju. Ocenka stojkosti parolej: sravnitel'noe issledovanie jeffektivnosti dliny i slozhnosti / A. Ju. Pravotorova // Nauka. Innovacii. budushhee – 2025 : Sbornik statej II Mezhdunarodnoj nauchno-prakticheskoj konferencii, Petrozavodsk, 15 maja 2025 goda. – Petrozavodsk: Mezhdunarodnyj centr nauchnogo partnerstva «Novaja Nauka» (IP Ivanovskaja I.I.), 2025. – S. 125–132. – EDN MEPXNB.
3. Khalil Nabab Pinjari. LSTM-Enabled Big Data Security Framework Integrating Kerberos Authentication on AWS for Robust Cloud Protection / Khalil Nabab Pinjari, Abu Zar Muhammad, Yogesh Kumar Sharma // Nanotechnology Perceptions. – 2024. – Vol. 20, No. 7. – DOI 10.62441/nano-ntp.v20i7.4414. – EDN ZHQLIB.
4. Abramova E. V. Vozmozhnosti Google Colab i Jupyter Notebook dlja reshenija zadach iskusstvennogo intellekta / E. V. Abramova, L. A. Maksimenko // Regulirovanie zemel'no-imushhestvennyh otnoshenij v Rossii: pravovoe i geoprostranstvennoe obespechenie, ocenka nedvizhimosti, jekologija, tehnologicheskie reshenija. – 2023. – № 1. – S. 23–29. – DOI 10.33764/2687-041X-2023-1-23-29. – EDN CYLSKT.
5. Il'ichev V. Ju. Analiz massivov dannyh s ispol'zovaniem biblioteki Pandas dlja Python / V. Ju. Il'ichev, E. A. Jurik // Nauchnoe obozrenie. Tehnicheskie nauki. – 2020. – № 4. – S. 41–45. – EDN BKGJHM.

4 Elena V. Bulgakova, Ph.D. in Law, associate professor, deputy chair of the department of information security for scientific research, Financial University under the Government of the Russian Federation, Moscow, Russia. E-mail: koordinator-proekta@mail.ru
5 Evgeny A. Bogdanov Ph.D., chair of the department of information security, Financial University under the Government of the Russian Federation, Moscow, Russia. E-mail: Eabogdanov@fa.ru
6 Alexander N. Kubankov, Dr.Sc. in Military Sciences, professor, professor of the department of information security, Financial University under the Government of the Russian Federation, Moscow, Russia. E mail: kan9991@gmail.com

6. Sudarikov G. V. Ispol'zovanie biblioteki Pandas dlja analiza dannyh / G. V. Sudarikov, I. A. Ashmarov // Mir obrazovanija - obrazovanie v mire. – 2023. – № 1(89). – S. 184–188. – DOI 10.51944/20738536_2023_1_184. – EDN NWHNRO.
7. Florescu D. A Machine Learning Based Software Pipeline to Pick the Variable Ordering for Algorithms with Polynomial Inputs / D. Florescu, M. England // Lecture Notes in Computer Science. – 2020. – Vol. 12097 LNCS. – P. 302–311. – DOI 10.1007/978-3-030-52200-1_30. – EDN JDDNYT.
8. Retnoningsih, E. Mengenal Machine Learning Dengan Teknik Supervised Dan Unsupervised Learning Menggunakan Python / E. Retnoningsih, R. Pramudita // Bina Insani ICT Journal. – 2020. – Vol. 7, No. 2. – P. 156. – DOI 10.51211/biict.v7i2.1422. – EDN TNZXLG.
9. Bulgakova E. V. Problema tochnosti i ob#jasnimosti pri vnedrenii iskusstvennogo intellekta v sistemy upravlenija informaciej i sobytijami bezopasnosti / E. V. Bulgakova, D. S. Dojnikov, A. N. Kubankov // Naukoemkie tehnologii v kosmicheskih issledovanijah Zemli. – 2025. – T. 17, № 3. – S. 35–41. – DOI 10.36724/2409-5419-2025-17-3-35-41. – EDN QBTNNT.
10. Magerramov I. M. Nechetkie podhody k resheniju zadach klassifikacii / I. M. Magerramov, T. S. Aleksandrova // Intellektual'nye resursy – regional'nomu razvitiju. – 2021. – № 2. – S. 76–82. – EDN XRMACX.
11. Feng, G. Common Python Data Analysis Method Based on Deep Learning / G. Feng // Journal of Physics: Conference Series. – 2021. – Vol. 2037, No. 1. – P. 012132. – DOI 10.1088/1742-6596/2037/1/012132. – EDN YQAYRU.
12. Abdulkadirov, R. Survey of Optimization Algorithms in Modern Neural Networks / R. Abdulkadirov, P. Lyakhov, N. Nagornov // Mathematics. – 2023. – Vol. 11, No. 11. – P. 2466. – DOI 10.3390/math11112466. – EDN BAUENY.
13. Tao, Ch. Applications of Bayesian Neural Networks in Outlier Detection / Ch. Tao // Big Data. – 2023. – Vol. 11, No. 5. – P. 369–386. – DOI 10.1089/big.2021.0343. – EDN DXZRFS.
14. Iiduka, H. Training deep neural networks using conjugate gradient-like methods / H. Iiduka, Y. Kobayashi // Electronics. – 2020. – Vol. 9, No. 11. – P. 1–25. – DOI 10.3390/electronics9111809. – EDN TMEEXV.
15. Trunov E. E. Obnaruzhenie ugroz bezopasnosti informacii s ispol'zovaniem glubokih nejronnyh setej v komp'juternyh setjah v rezhime real'nogo vremeni / E. E. Trunov, S. G. Kljuev // Modelirovanie, optimizacija i informacionnye tehnologii. – 2022. – T. 10, № 3(38). – S. 12-13. – DOI 10.26102/2310-6018/2022.38.3.011. – EDN MNLGVN.
16. Nakhushev, R. S. Application of the neural networks for cryptographic information security / R. S. Nakhushev, N. V. Sukhanova // Proceedings of the 2020 IEEE International Conference «Quality Management, Transport and Information Security, Information Technologies», IT and QM and IS 2020, Yaroslavl, 07–11 sentjabrja 2020 goda. – Yaroslavl, 2020. – P. 421–423. – DOI 10.1109/ITQMIS51053.2020.9322981. – EDN FALTNR.