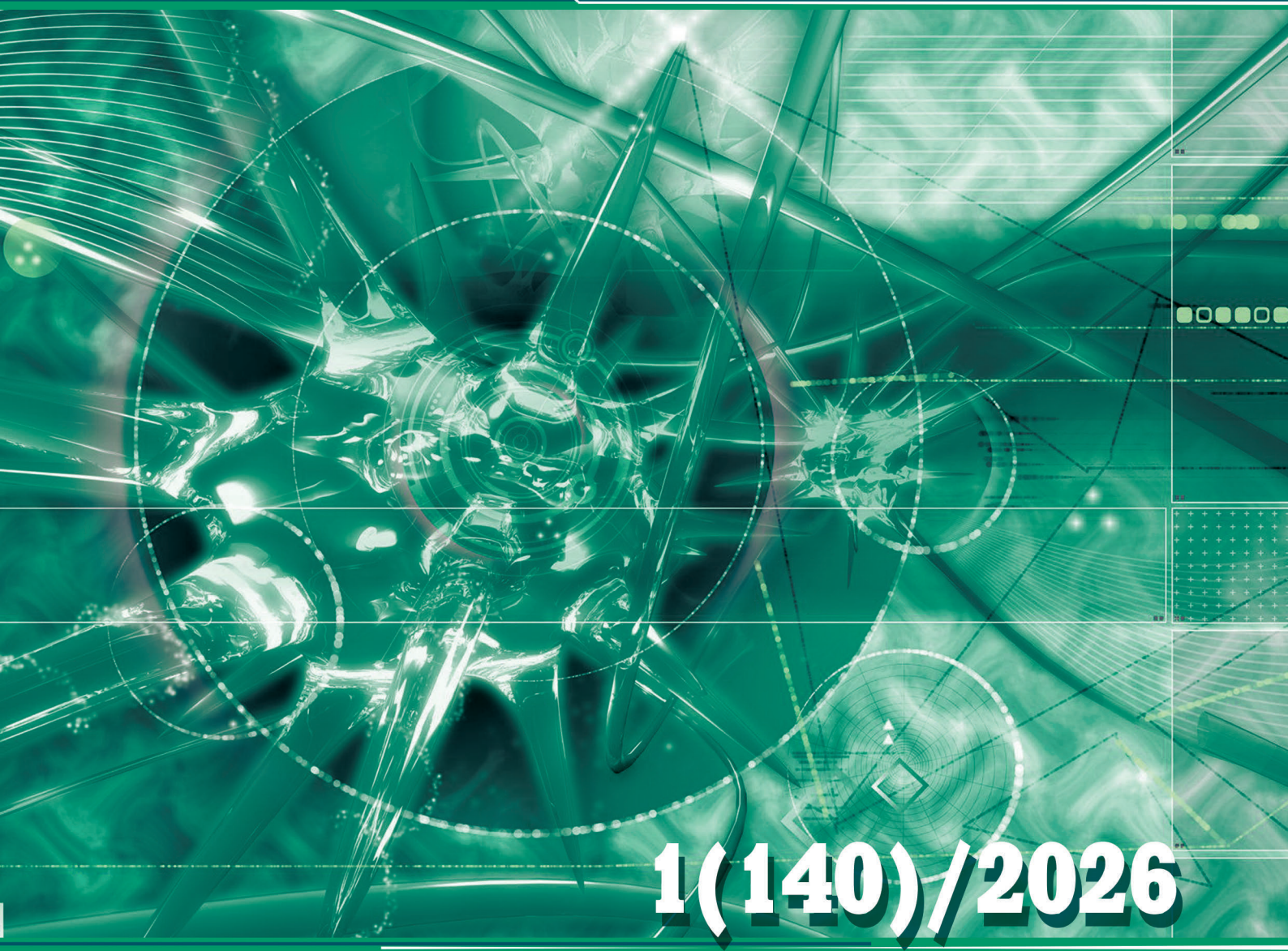
ISSN 1684-8853 (print); ISSN 2541-8610 (online)

ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ СИСТЕМЫ

НАУЧНЫЙ ЖУРНАЛ



1(140)/2026

doi:10.31799/1684-8853-2026-1

ISSN 1684-8853 (print); ISSN 2541-8610 (online)

SCIENTIFIC JOURNAL

1(140)/2026

PEER REVIEWED JOURNAL

INFORMATSIONNO-UPRAVLIAIUSHCHIE SISTEMY (INFORMATION AND CONTROL SYSTEMS)

Contact information
The Editorial and Publishing Center, SUAI
67A, Bol'shaya Morskaia, 190000, Saint Petersburg, Russia
Website: http://i-us.ru/en, e-mail: ius.spb@gmail.com
Tel.: +7 - 812 494 70 02

The Journal was registered in the Ministry of Press, Broadcasting and Mass Media of the Russian Federation.
Registration Certificate JD № 77-12412 from April, 19, 2002.
Re-registration in the Federal Service for Supervision in the Sphere of Telecom, Information Technologies and Mass Communications (ROSKOMNADZOR) due to change of the founder A. Vostrikov,
PI № FS 77-82226 from November, 23, 2021

Submitted for publication 09.01.26. Passed for printing 25.02.26. Signed for printing: 27.02.26.
Format $60\times84_{1/8}$. Phototype CentSchbkCyrill BT. Digital printing.

Layout original is made at the Editorial and Publishing Center, SUAI.
67A, Bol'shaya Morskaia, 190000, Saint Petersburg, Russia
Printed from slides at the Editorial and Publishing Center, SUAI.
67A, Bol'shaya Morskaia, 190000, Saint Petersburg, Russia

The journal is indexed in Scopus.

Free distribution.

doi:10.31799/1684-8853-2026-1

ISSN 1684-8853 (print); ISSN 2541-8610 (online)

НАУЧНЫЙ ЖУРНАЛ

1(140)/2026

РЕЦЕНЗИРУЕМОЕ ИЗДАНИЕ

ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ СИСТЕМЫ

Адрес редакции: 190000, г. Санкт-Петербург,
ул. Большая Морская, д. 67, лит. А, ГУАП, РИЦ
Тел.: (812) 494-70-02, эл. адрес: ius.spb@gmail.com,
сайт: http://i-us.ru

Журнал входит в БД Scopus и в Перечень рецензируемых научных изданий, в которых должны быть опубликованы основные научные результаты диссертаций на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук.

Сдано в набор 09.01.26. Подписано в печать 25.02.26. Дата выхода в свет: 27.02.26.
Формат 60×841/8. Гарнитура CentSchbkCyrill BT. Печать цифровая.
Усл. печ. л. 10,8. Уч.-изд. л. 14,8. Тираж 1000 экз (1-й завод 50 экз.). Заказ № 35.

Оригинал-макет изготовлен в редакционно-издательском центре ГУАП.
190000, г. Санкт-Петербург, ул. Большая Морская, д. 67, лит. А.

Отпечатано в редакционно-издательском центре ГУАП.
190000, г. Санкт-Петербург, ул. Большая Морская, д. 67, лит. А.

Распространяется бесплатно.

Журнал зарегистрирован в Министерстве РФ по делам печати, телерадиовещания и средств массовых коммуникаций.
Свидетельство о регистрации ПИ № 77-12412 от 19 апреля 2002 г.
Перерегистрирован в Роскомнадзоре.
Свидетельство о регистрации ПИ № ФС77-82226 от 23 ноября 2021 г.

УДК 519.614
doi:10.31799/1684-8853-2026-1-2-7
EDN: OANKSB

Научные статьи
Articles

Памяти моего учителя и соавтора
профессора Балонина Николая Алексеевича

Взаимосвязь симметрий бициклических ортогональных матриц и их порядков

***А. М. Сергеев*[a],** *канд. техн. наук, доцент, orcid.org/0000-0002-4788-9869, aleks.asklab@gmail.com*
[a]*Санкт-Петербургский государственный университет аэрокосмического приборостроения, Б. Морская ул., 67, Санкт-Петербург, 190000, РФ*

Введение: *матричные вычисления, являясь структурированными и простыми, применяются в различных задачах и технических системах, включая криптографические, телекоммуникационные и др. Особый интерес для практического применения представляют ортогональные матрицы Адамара и конференц-матрицы Белевича с различными симметриями, исследование которых редко распространяется на их блочные структуры.* ***Цель:*** *показать способы получения симметричных матриц семейства Адамара бициклических структур с окаймлением, понимая симметрию в широком смысле, включая кососимметрию и двоякосимметрию.* ***Результаты:*** *выявлены взаимосвязь симметрий бициклических матриц с каймой с их порядками, равными простым числам и степеням простых чисел, а также способы их получения на основе симметричных и кососимметричных блоков, позволяющие расширить представительство матриц на указанных порядках и возможность их выбора для конкретного применения.* ***Обсуждение:*** *симметрия в ортогональных матрицах представляет собой малоизученное явление, особенно для блочных структур таких матриц, хотя имеет существенное значение для их практических применений. Интерес вызывает исследование условий существования бициклических ортогональных матриц с каймой (двойной каймой), состоящих из пары циклических блоков — кососимметричного и симметричного.*

Ключевые слова *— ортогональные матрицы, бициклические матрицы, симметрия, кососимметрия, двоякая симметрия.*

Для цитирования: Сергеев А. М. Взаимосвязь симметрий бициклических ортогональных матриц и их порядков. *Информационно-управляющие системы*, 2026, № 1, с. 2–7. doi:10.31799/1684-8853-2026-1-2-7, EDN: OANKSB
For citation: Sergeev A. M. The relationship of bicyclic orthogonal matrices symmetries and their orders. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2026, no. 1, pp. 2–7 (In Russian). doi:10.31799/1684-8853-2026-1-2-7, EDN: OANKSB

Введение

Ортогональные матрицы с ограниченным числом целочисленных значений элементов привлекают внимание разработчиков методов и средств, широко используемых, например, в каналах открытых коммуникационных систем для помехозащищенного и помехоустойчивого преобразования данных [1], при формировании преамбул сообщений, передаваемых по каналам связи [2], в кодировании [3], криптографии [4] и др.

Такими матрицами являются квадратные ортогональные матрицы Адамара $\mathbf{H}$ с элементами $\{1, –1\}$, существующие на четных порядках n, для которых справедливо $\mathbf{H}^{\mathrm{T}}\mathbf{H} = n\mathbf{I}$ [5]. Здесь $n = 4t$, где t — натуральное число, а $\mathbf{I}$ — единичная матрица. На четных порядках $n = 4t – 2$ существуют матрицы Белевича $\mathbf{C}$ с элементами $\{1, 0, –1\}$ [6, 7].

Большой научный и практический интерес представляют для указанных задач такие матрицы с симметричными структурами или симметриями в структуре и способы их поиска либо конструирования.

Из линейной алгебры о симметрии квадратных матриц известно, что:

— для любой матрицы $\mathbf{A}$ матрицы $\mathbf{A}\mathbf{A}^{\mathrm{T}}$, $\mathbf{A}^{\mathrm{T}}\mathbf{A}$ и $\mathbf{A} + \mathbf{A}^{\mathrm{T}}$ являются симметричными;

— для любых матриц $\mathbf{A}$ и $\mathbf{B}$ произведение $\mathbf{AB}$ — симметричная матрица, если справедливо $\mathbf{AB} = \mathbf{BA}$.

Однако приведенные действия даже с ортогональными матрицами $\mathbf{A}$ и $\mathbf{B}$ не могут быть использованы для поиска симметричных ортогональных матриц $\mathbf{H}$, поскольку при их выполнении результирующая симметричная матрица теряет ортогональность.

Теория матриц семейства Адамара в последнее время значительно расширилась представлениями о связях порядков их существования с числовыми последовательностями и симметрией структур [8, 9].

Симметрия — это свойство, известное лишь отчасти в отношении ортогональных матриц, поэтому цель статьи — дать о них представление, сформированное относительно недавно, и показать метод формирования матриц с симметриями на основе бициклических структур (би-

циклов), понимая симметрию в широком смысле: это симметрия, кососимметрия [10] и двоякая симметрия [6].

Матрицы Одина семейства Адамара

Можно предположить, что симметрия может проявляться для матриц Адамара различной структуры. Однако здесь ограничимся рассмотрением симметричных матриц, получаемых на основе циклических структур.

Поскольку, согласно гипотезе Райзера [11], симметричных матриц Адамара циклической конструкции выше четвертого порядка нет, то будем рассматривать бициклические матрицы с каймой [12], построенные на основе двух циклических блоков **A** и **B**, — матрицы Одина [13].

В семействе Адамара есть матрицы, безразличные к простоте числа $n-1$, связанного с их порядком $n = 4t$. Вычитание 1 определяет то, что свойства матрицы связаны с основой (ядром) [13] нечетного порядка и добавленной к ней каймы — строки сверху и столбца слева.

Есть также и зависящие от свойств $n-1$ основы, причем на порядках $n = 4t-2$ матрицам Адамара противопоставляют матрицы Белевича с нулевой диагональю diag{0, 0, 0, … 0} [14].

Различие между матрицами Адамара и Белевича состоит в том, что последние представлены на обеих последовательностях четных порядков $4t$ и $4t-2$. Основу матрицы Белевича порядков $4t-2$ можно подать в виде симметричной конструкции, состоящей из бицикла с бинарной каймой порядка $4t-3$.

Приведем определения матриц Одина порядков $4t-1$ и $4t-3$.

«*Определение 1.* Матрица Одина — это квадратная матрица порядка $4t-1$, являющегося простым числом или его степенью, со значениями элементов 1, $-b$ и $d = 0$ на диагонали, где $b = \dfrac{v-1}{v+\sqrt{2v-1}}$, $v = (n-1)/2$ — половина порядка матрицы, без учета ее каймы d.

Определение 2. Матрица Одина порядка $4t-3$, являющегося простым числом или его степенью, — это квадратная матрица со значениями элементов 1, $-b$, $d = \dfrac{1}{1+\sqrt{n}}$ (на диагонали), где $b = 1-2d$» [13].

Инвариантом матрицы Одина является матрица с равным числом внедиагональных элементов с одинаковым значением. Такая структура позволяет с легкостью выделить в качестве первой строки и столбца кайму из элементов векторов **e** и $-b\mathbf{e}$, где **e** — единичный вектор длины v.

Матрицы описываются соответственно кососимметричной и симметричной структурами:

$$\mathbf{O}_{4t-1} = \begin{pmatrix} d & \mathbf{e} & -b\mathbf{e} \\ -b\mathbf{e} & \mathbf{A} & \mathbf{B} \\ \mathbf{e} & \left[-\mathbf{B}^{\mathrm{T}}\right] & \mathbf{A}^{\mathrm{T}} \end{pmatrix};$$

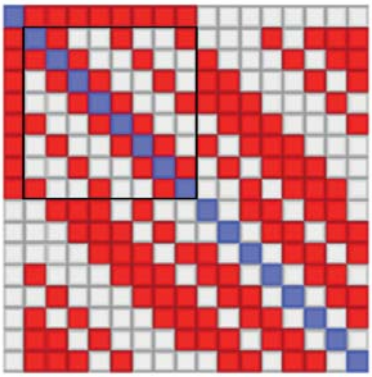
$$\mathbf{O}_{4t-3} = \begin{pmatrix} d & -b\mathbf{e} & \mathbf{e} \\ -b\mathbf{e} & \mathbf{A} & \mathbf{B} \\ \mathbf{e} & \mathbf{B}^{\mathrm{T}} & \left[-\mathbf{A}^{\mathrm{T}}\right] \end{pmatrix}.$$

В приведенных выше структурах [•] обозначает процедуру замены всех положительных элементов на 1, а отрицательных — на $-b$. Добавление верхней строки из 1 и левого столбца из –1 к матрице порождает матрицу Белевича с элементами $-b = -1$, $d = 0$.

Портрет матрицы Одина порядка 17 представлен на рис. 1. Здесь элементы матрицы с отрицательными значениями отмечены клетками красного цвета, элементы с положительными значениями — белого цвета. Диагональные элементы матриц Одина обычно нормируются так, чтобы вернуть основе ортогональность, — они синего цвета.

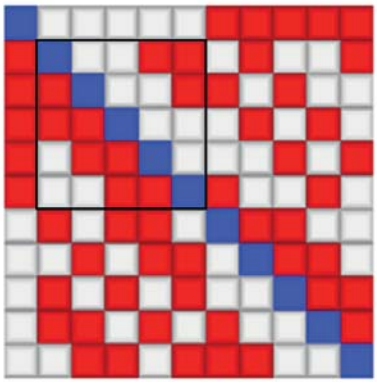
У этой матрицы как основы матрицы Адамара есть свойство, позволяющее ее находить. Блок **A** бицикла, как видно, симметричен, а блок **B** — почти кососимметричен, т. е. первая строка состоит из инвертированных по знаку и по расположению элементов [15].

Характерно, что этот вид симметрии существует только на порядках $4t-3$, на которых есть поля Галуа GF(n). Поэтому двоякосимметричные матрицы Одина существуют только на порядках, равных простым числам и степеням простых чисел.



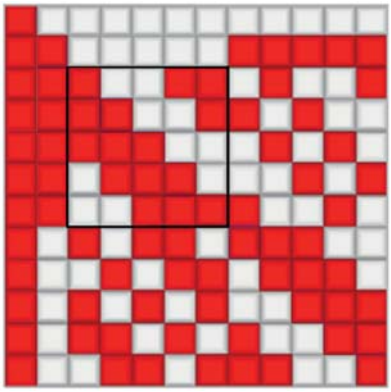
■ ***Рис. 1.*** Портрет матрицы Одина порядка 17 с выделенным блоком **A**

■ ***Fig. 1.*** Portrait of Odin matrix of order 17 with a dedicated block **A**



■ ***Рис. 2.*** Портрет матрицы Одина порядка 11 с выделенным блоком **A**

■ ***Fig. 2.*** Portrait of Odin matrix of order 11 with a dedicated block **A**



■ ***Рис. 3.*** Портрет матрицы Адамара порядка 12 как результат окаймления матрицы Одина порядка 11

■ ***Fig. 3.*** Portrait of the Hadamard matrix of order 12 as a result of adding a border to the Odin matrix of order 11

Известны рудименты в виде симметричных в целом матриц **C** на порядках, равных числам, разлагаемым на сумму квадратов двух чисел, но при этом они настолько быстро усложняются структурно [16], что их не умеют находить (конструировать) уже на порядках 66 и 86. Конструкция матрицы Белевича на порядке 46 и приемы ее построения оказались неприменимы для более высоких порядков.

Все это выделяет двоякосимметричные структуры в особую разновидность, сопровождающую простые числа. Разумеется, появления аналогичных структур следует ожидать на порядках $n = 4t - 1$, порядках кососимметричных основ матриц Адамара. Портрет такой основы порядка 11 приведен на рис. 2.

У этой основы, как видно, блок **A** бицикла кососимметричен, блок **B** — симметричен, дефект кососимметрии [15] отсутствует. Это означает, что матрица Адамара как результат окаймления матрицы Одина порядка 11, портрет которой представлен на рис. 3, может быть после соответствующей перестановки половины ее столбцов как кососимметричной, так и симметричной — это двоякосимметричная матрица.

Вычисление элементов матриц Одина не изменяется, но если диагональные элементы принимают значение 1, а это возможно только для кососимметричных версий, то это квазиортогональные матрицы Мерсенна [6].

Матрицы Мерсенна и Эйлера семейства Адамара

В отличие от матриц Одина, матрицы Мерсенна обладают качеством, резко выделяющим матрицы Адамара, построенные на их основе. Воспользуемся известными определениями матриц Мерсенна и Эйлера.

«*Определение 3.* Матрица Мерсенна **M** — квадратная матрица порядка $n = 4t - 1$ со значениями элементов 1 и $-b$, столбцы которой ортогональны $\mathbf{M}^{\mathrm{T}}\mathbf{M} = \mu\mathbf{I}$, $b = \dfrac{t}{t+\sqrt{t}}$, $\mu = \dfrac{p+qb^2}{2}$, $p = n - 1$, $q = n + 1$ (порядок матрицы Адамара).

Определение 4. Матрица Эйлера **E** — квадратная матрица порядка $n = 4t - 2$ со значениями элементов $1, -a, b, -b$, столбцы которой ортогональны $\mathbf{E}^{\mathrm{T}}\mathbf{E} = \xi\mathbf{I}$, где $b = 1/2$ при $n = 6$, в остальных случаях $b = \dfrac{q-\sqrt{8q}}{q-8}$, $q = n + 2$ (порядок матрицы Адамара), вес $\xi = \dfrac{(n+2)+(n-2)b^2}{2}$ учитывает, что $q/2$ элементов каждого столбца такой матрицы имеют значения $|a| = 1$, модули остальных элементов равны $|b| < 1$» [6].

Матрицы Эйлера, как и матрицы Адамара, можно вычислять по правилу Сильвестра, используя матрицы Мерсенна вдвое меньшего порядка. Это правило является общим для всех адамаровых матриц и представляется в виде [16]

$$\mathbf{E}_n = \begin{pmatrix} \mathbf{M}_{n/2} & \mathbf{M}_{n/2} \\ \mathbf{M}_{n/2} & -\mathbf{M}_{n/2} \end{pmatrix}.$$

В то же время матрицы Мерсенна связаны с матрицами Эйлера дополнением их строкой и столбцом (каймой) в виде [5]

$$\mathbf{M}_{n+1} = \begin{pmatrix} -\lambda & \mathbf{e}^{\mathrm{T}} \\ \mathbf{e} & \mathbf{E}_n^{*} \end{pmatrix},$$

где $\lambda = -a$ — собственное число, а **e** — собственный вектор «сопряженной» матрицы

$$\mathbf{E}_n^* = \begin{pmatrix} \mathbf{M}_{n/2} & \mathbf{M}_{n/2} \\ \mathbf{M}_{n/2} & \mathbf{M}_{n/2}^* \end{pmatrix},$$

блок $\mathbf{M}_{n/2}^*$ получается из $\mathbf{M}_{n/2}$ взаимной заменой элементов 1 и $-b$ и пересчетом уровня $b = \dfrac{q - \sqrt{4q}}{q - 4}$, где $q = n + 2$ (порядок матрицы Адамара).

Матрицы Адамара являются ограниченно возрастающими по сложности матрицами в том смысле, что для существования их на любом порядке $4t$ достаточно платы в виде потери обоих видов симметрий двумя блоками ее бицикла.

Альтернативная формулировка гипотезы Адамара выглядит следующим образом: нет такой матрицы Адамара, для которой не нашелся бы бицикл Эйлера. Название основы — от основы, получаемой отделением второй каймы.

Матрицы Эйлера похожи на матрицы Адамара — они существуют на четных порядках и также могут быть представлены двумя блоками в виде

$$\mathbf{E}_{2n} = \begin{pmatrix} \mathbf{A}_n & \mathbf{B}_n \\ \mathbf{B}_n^{\mathrm{T}} & -\mathbf{A}_n^{\mathrm{T}} \end{pmatrix}.$$

Бицикл Эйлера максимально прост по своей конструкции, число положительных элементов в его блоках на единицу превышает число отрицательных, и это важнейший инвариант матриц Адамара после приведения ее к основе удалением парной каймы и замены элементов.

За этой стойкостью стоит несомненный математический факт, что если матрицы Одина сопровождают простые числа и степени простых чисел, то матрицы Эйлера сопровождают все, без исключения, числа $4t - 1$. Для того чтобы не существовало матрицы Эйлера, надо, чтобы целое число не было простым или составным, входящим в эту последовательность. Система целых чисел ограничена в своей сложности всего этими двумя типами, третьего нет, а матрица Эйлера не может то существовать, то не существовать по своему усмотрению. Это минимальное доказательство гипотезы Адамара, построенное на интерпретации принципа сложности [17].

Простые числа неизменно обнаруживаются в любом количестве, что доказано еще в античные времена. Следовательно, число матриц Одина бесконечно. Также бесконечно число матриц Эйлера, безразличных к потере свойства парной симметрии.

Верхняя строка блока **B** антисимметрична, она инвертирована и меняет знак или же она просто симметрична. Это напоминает синусы и косинусы. Неудивительно, что обнаруженные двоякосимметричные матрицы сопровождают порядки, где есть поля Галуа. Ведь функция полей — описывать симметрии. Таким образом, у простых чисел и их степеней есть сопровождающие их матрицы. Причем расчет в сложных полях дает матрицы с тем же узором портрета. Из этого можно сделать вывод, что кососимметрично-симметричного решения за пределами этой области нет. Нет полей, нет расчета, нет матриц.

Таким образом, разбор зависимости симметрий матрицы Одина от порядка показывает, что для $n = 4t - 3$ матрица является симметричной, а для $n = 4t - 1$ матрица остается симметричной или становится кососимметричной.

Заключение

Кососимметричная форма предпочтительна для построения матриц Мерсенна. Но это неполная ее характеристика, поскольку поля связаны не с одинарной, а с парной симметрией. Парносимметрична также и сама бициклическая матрица Эйлера. У матриц Белевича бициклическая основа именуется матрицей тени, под этим имеется в виду, что это структурная тень самих матриц Белевича.

На основе использования принципа сложности можно построить вполне состоятельную теорию ортогональных матриц, поясняющую причины, по которым некоторые матрицы Белевича не найдены и не будут найдены никогда, поскольку сложность узора их портретов возрастает по обоим измерениям.

Матрицы Эйлера существуют на более высоких порядках, чем матрицы тени, и их сложность одномерна — касается только одного измерения, а именно двух верхних строк блоков **A** и **B** бициклических структур. Если строки двоякосимметричны, их находят полями Галуа [18], в противном случае приходится применять более сложные подходы, основанные на конструировании по требованиям дизайна [19].

Финансовая поддержка

Работа выполнена при финансовой поддержке Министерства науки и высшего образования Российской Федерации, соглашение № FSRF-2023-0003 «Фундаментальные основы построения помехозащищенных систем космической и спутниковой связи, относительной навигации, технического зрения и аэрокосмического мониторинга».

Литература

1. **Хвощ С. Т.** Об особенностях реализации помехозащищенного кодирования изображений. *Вопросы радиоэлектроники. Серия: Техника телевидения*, 2024, № 4, с. 60–65. EDN: MBZPNF
2. **Чистяков Е. А., Мартынов И. А., Самохина Е. В.** Кодовое разделение каналов. *Вопросы электромеханики. Труды ВНИИЭМ*, 2023, т. 192, № 1, с. 27–32.
3. **Chathely B. J.** Hadamard matrix and its application in coding theory and com-binatorial design theory. *International Journal of Mathematics Trends and Technology*, 2018, vol. 59, iss. 4, pp. 218–227. doi:10.14445/22315373/IJMTT-V59P532
4. *New Advances in Designs, Codes and Cryptography*. Ch. J. Colbourn, J. H. Dinitz (eds), Stinson66, Toronto, Canada, June 13–17, 2022. Switzerland, Cham, 2022. 425 p. https://doi.org/10.1007/978-3-031-48679-1
5. **Jennifer S., Yamada M.** *Hadamard Matrices: Constructions using Number Theory and Linear Algebra*. Wiley, 2020. 384 p.
6. **Балонин Н. А., Сергеев М. Б.** *Специальные матрицы: псевдообратные, ортогональные, адамаровы и критские*: монография. СПб., Политехника, 2019. 196 с. doi:10.25960/7325-1155-0
7. **Goethals J. M., Seidel J. J.** Orthogonal matrices with zero diagonal. *Canadian Journal of Mathematics*, 1967, no. 19, pp. 1001–1010.
8. **Balonin N. A., Jennifer Seberry.** A review and new symmetric conference matrices. *Информационно-управляющие системы*, 2014, № 4, с. 2–7.
9. **Ang M. H., Ma S. L.** Symmetric Weighing matrices constructed using group matrices. *Design, Codes and Cryptography*, 2005, vol. 37, pp. 195–210.
10. **Kravvaritis C., Mitrouli M., Jennifer S.** On the growth problem for skew and symmetric conference matrices. *Linear Algebra and its Applications*, 2005, vol. 403, pp. 183–206.
11. **Балонин Н. А., Сергеев М. Б.** Расширение гипотезы Райзера на двуциклические структуры и разрешимость матриц Адамара орнаментом в виде бицикла с двойной каймой. *Информационно-управляющие системы,* 2017, № 1, с. 2–10. doi:10.15217/issn1684-8853.2017.1.2
12. **Балонин Н. А., Джокович Д. Ж.** Симметрия двуциклических матриц Адамара и периодические пары Голея. *Информационно-управляющие системы*, 2015, № 3, с. 2–16. doi:10.15217/issn1684-8853.2015.3.2
13. **Балонин Н. А., Сергеев М. Б.** Критские матрицы Одина и Тени, сопровождающие простые числа и их степени. *Информационно-управляющие системы,* 2022, № 1, с. 2–7. doi:10.31799/1684-8853-2022-1-2-7
14. **Belevitch V.** Theorem of 2n-terminal networks with application to conference telephony. *Electrical Communications*, 1950, vol. 26, pp. 231–244.
15. **Sergeev A., Sergeev M., Balonin N., Vostrikov A.** Symmetry indices as a key to finding matrices of cyclic structure for noise-immune coding. *Smart Innovation, Systems and Technologies,* 2020, vol. 193, pp. 223–230. doi:10.1007/978-981-15-5925-9_19
16. **Balonin N. A., Vostrikov A. A., Sergeev M. B.** On two predictors of calculable chains of quasi-orthogonal matrices. *Automatic Control and Computer Sciences*, 2015, vol. 49, no. 3, pp. 153–158.
17. **Балонин Н. А., Сергеев А. М.** *Порядок и беспорядок в мире матриц, принцип неограниченно возрастающей сложности*. Математические методы и модели в высокотехнологичном производстве: сб. тезисов докладов II Международного форума, Санкт-Петербург, 2022, с. 14–17.
18. **Балонин Н. А., Сергеев А. М., Синицына О. А.** Алгоритмы конечных полей и групп поиска ортогональных последовательностей. *Информационно-управляющие системы,* 2021, № 4, с. 2–17. doi:10.31799/1684-8853-2021-4-2-17
19. **Colbourn C. J., Dinitz J. H.** *Handbook of Combinatorial Designs*. Second Ed. Chapman and Hall/CRC, 2007. 967 p.

UDC 519.614
doi:10.31799/1684-8853-2026-1-2-7
EDN: OANKSB

In memory of Professor Nikolai Balonin, my teacher and co-author

The relationship of bicyclic orthogonal matrices symmetries and their orders

A. M. Sergeev[a], PhD, Tech., Associate Professor, orcid.org/0000-0002-4788-9869, aleks.asklab@gmail.com
[a]Saint-Petersburg State University of Aerospace Instrumentation, 67, B. Morskaia St., 190000, Saint-Petersburg, Russian Federation

Introduction: Matrix calculations, being structured and simple, are used in various tasks and technical systems, including cryptographic systems, telecommunication ones, etc. Of particular interest for practical application are orthogonal Hadamard matrices and Belevich conference matrices with various symmetries, the study of which rarely extends to their block structures. **Purpose:** To show the ways of obtaining symmetric matrices of the Hadamard family of bicyclic structures with borders , using symmetry in a broad sense,

including skew symmetry and dual symmetry. **Results:** We demonstrate the relationship of symmetries of bicyclic matrices with a border with their orders equal to primes and powers of primes. We also show the ways to obtain them based on symmetric and skew-symmetric blocks, which allow expanding the representation of matrices on these orders and the possibility of their choice for a specific application. **Discussion:** Symmetry in orthogonal matrices is a little-studied phenomenon, especially for block structures of such matrices, although it is essential for their practical matrix applications. It is of interest to study the conditions for the existence of bicyclic orthogonal matrices with a border (double border) consisting of a pair of cyclic blocks – skew-symmetric and symmetric ones.

Keywords — orthogonal matrices, bicyclic matrices, symmetry, skew symmetry, two-fold symmetry.

For citation: Sergeev A. M. The relationship of bicyclic orthogonal matrices symmetries and their orders. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2026, no. 1, pp. 2–7 (In Russian). doi:10.31799/1684-8853-2026-1-2-7, EDN: OANKSB

Financial support

The work was carried out with the financial support of the Ministry of Science and Higher Education of the Russian Federation, agreement No. FSRF-2023-0003 "Fundamental principles for constructing interference-resistant systems of space and satellite communications, relative navigation, machine vision and aerospace monitoring".

References

1. Hvoshch S. T. About the features of the implementation of noise-proof image encoding. *Voprosy radioelektroniki. Seriya: Tekhnika televideniya*, 2024, no. 4, pp. 60–65 (In Russian). EDN: MBZPNF
2. Chistyakov E. A., Martynov I. A., Samohina E. V. Code division of channels. *Voprosy elektromekhaniki. Trudy VNIIEM*, 2023, vol. 192, no. 1, pp. 27–32 (In Russian). EDN: OUOJYB
3. Chathely B. J. Hadamard matrix and its application in coding theory and com-binatorial design theory. *International Journal of Mathematics Trends and Technology*, 2018, vol. 59, iss. 4, pp. 218–227. doi:10.14445/22315373/IJMTT-V59P532
4. *New Advances in Designs, Codes and Cryptography*. Ch. J. Colbourn, J. H. Dinitz (eds), Stinson66, Toronto, Canada, June 13–17, 2022. Switzerland, Cham, 2022. 425 p. https://doi.org/10.1007/978-3-031-48679-1
5. Jennifer S., Yamada M. *Hadamard Matrices: Constructions using Number Theory and Linear Algebra*. Wiley, 2020. 384 p.
6. Balonin N. A., Sergeev M. B. *Special`ny`e matricy: psevdo-obratny`e, ortogonal`ny`e, adamarovy` i kritskie* [Special matrices: pseudo-return, orthogonal, Hadamardian and Cretan]. Saint-Petersburg, Politekhnika Publ., 2019. 196 p. (In Russian) https://doi.org/10.25960/7325-1155-07
7. Goethals J. M., Seidel J. J. Orthogonal matrices with zero diagonal. *Canadian Journal of Mathematics*, 1967, no. 19, pp. 1001–1010.
8. Balonin N. A., Jennifer Seberry. A review and new symmetric conference matrices. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2014, no. 4, pp. 2–7.
9. Ang M. H., Ma S. L. Symmetric Weighing matrices constructed using group matrices. *Design, Codes and Cryptography*, 2005, vol. 37, pp. 195–210.
10. Kravvaritis C., Mitrouli M., Jennifer S. On the growth problem for skew and symmetric conference matrices. *Linear Algebra and its Applications*, 2005, vol. 403, pp. 183–206.
11. Balonin N. A., Sergeev M. B. Ryser's conjecture expansion for bicirculant strictures and Hadamard matrix resolvability by double-border bicycle ornament. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2017, no. 1, pp. 2–10 (In Russian). doi:10.15217/issn1684-8853.2017.1.2
12. Balonin N. A., Djokovic D. Z. Symmetry of two-circulant Hadamard matrices and periodic Golay pairs. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2015, no. 3, pp. 2–16 (In Russian). doi:10.15217/issn1684-8853.2015.3.2
13. Balonin N. A., Sergeev M. B. Odin and Shadow Cretan matrices accompanying primes and their powers. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2022, no. 1, pp. 2–7 (In Russian). doi:10.31799/1684-8853-2022-1-2-7
14. Belevitch V. Theorem of 2n-terminal networks with application to conference telephony. *Electrical Communications*, 1950, vol. 26, pp. 231–244.
15. Sergeev A., Sergeev M., Balonin N., Vostrikov A. Symmetry indices as a key to finding matrices of cyclic structure for noise-immune coding. *Smart Innovation, Systems and Technologies*, 2020, vol. 193, pp. 223–230. doi:10.1007/978-981-15-5925-9_19
16. Balonin N. A., Vostrikov A. A., Sergeev M. B. On two predictors of calculable chains of quasi-orthogonal matrices. *Automatic Control and Computer Sciences*, 2015, vol. 49, no. 3, pp. 153–158.
17. Balonin N. A., Sergeev A. M. Order and disorder in the world of matrices, the principle of infinitely increasing complexity. *Trudy II Mezhdunarodnogo Foruma "Matematicheskie metody i modeli v vysokotekhnologichnom proizvodstve"* [Proc. II International Forum "Mathematical methods and models in high-tech production"]. Saint-Petersburg, 2022, pp. 14–17 (In Russian).
18. Balonin N. A., Sergeev A. M., Sinitsyna O. I. Finite field and group algorithms for orthogonal sequence search. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2021, no. 4, pp. 2–17 (In Russian). doi:10.31799/1684-8853-2021-4-2-17
19. Colbourn C. J., Dinitz J. H. *Handbook of Combinatorial Designs*. Second Ed. Chapman and Hall/CRC, 2007. 967 p.

УДК 519.6
doi:10.31799/1684-8853-2026-1-8-18
EDN: QIENHD

Научные статьи
Articles

Проблема CSM в частотной области изображения: новый подход к решению

Р. А. Солодуха[a]***,*** *канд. техн. наук, доцент, /orcid.org/0000-0002-3878-4221, standartal@list.ru*
[a]*Воронежский государственный университет инженерных технологий, Революции пр., 19, Воронеж, 394036, РФ*

Введение: *одной из проблем, препятствующих применению стеганоанализа в практике цифровой криминалистики, является несоответствие тестируемого контейнера множеству, на котором обучалась модель распознавания. Имеющиеся методы решения данной проблемы не учитывают специфику искажений контейнера, привносимых различными стеганоалгоритмами. При определенной локализации искажений возможно их использование для формирования обучающего множества, соответствующего тестируемому контейнеру.* ***Цель:*** *формализация и проверка гипотезы об эффективности формирования обучающего множества на основе упорядочивания расстояния между «калиброванными изображениями» по векторам стеганоаналитических признаков при локализации искажений в частотной области изображений; сравнение точности распознавания при предлагаемом подходе и объединении признаков «калиброванного изображения» и исходного в единый вектор признаков.* ***Результаты:*** *показана целесообразность использования стеганоаналитических векторов признаков «калиброванных изображений» для вычисления расстояния между контейнерами. Предложена формализованная процедура формирования обучающего множества на основе расстояния между контейнерами. Создана программная инфраструктура для проведения численного эксперимента. Экспериментально показано, что независимо от качества JPEG-сжатия использование вектора признаков «калиброванного изображения» для формирования обучающего множества эффективнее, чем включение в общий вектор признаков. При этом основные вычисления задействованы на попарный расчет расстояния между контейнерами и могут осуществляться до появления объекта исследования. Для обеспечения воспроизводимости эксперимента наборы данных и программный код представлены в Kaggle.* ***Практическая значимость:*** *на примере стеганоалгоритма nsF5 показано преимущество применения стеганоаналитического вектора PEV-274 на соответствующем тестируемому файлу обучающем множестве перед CC-PEV-548 на случайной выборке. Предложенный подход способствует как увеличению точности стеганоанализа, так и уменьшению сроков исследования, что важно в экспертной практике.*

Ключевые слова *— стеганоанализ, вектор признаков, nsF5, PEV-274, CC-PEV-548, Cover-Source Mismatch, стеганография, машинное обучение, регрессия, расстояние между векторами, экспертиза.*

Для цитирования: Солодуха Р. А. Проблема CSM в частотной области изображения: новый подход к решению. *Информационно-управляющие системы*, 2026, № 1, с. 8–18. doi:10.31799/1684-8853-2026-1-8-18, EDN: QIENHD
For citation: Solodukha R. A. A novel approach to solving the CSM problem in the frequency domain of an image. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2026, no. 1, pp. 8–18 (In Russian). doi:10.31799/1684-8853-2026-1-8-18, EDN: QIENHD

Введение

Тенденции последних конкурсов по стеганоанализу Alaska [1] и Alaska-2 [2] демонстрируют интерес исследователей к наборам данных, содержащих полноцветные изображения, полученные с различных устройств. Основное внимание уделяется точности обнаружения при минимальном количестве ложных срабатываний. Это, а также появление теоретических работ по стеганоаналитической экспертизе [3] свидетельствует о том, что выявление стеганографии в цифровых изображениях находится на пороге качественного скачка. Вероятно, в среднесрочной перспективе стеганоанализ перейдет из исследовательских лабораторий и CTF в практическую сферу цифровой криминалистики.

За три десятилетия развития стеганоанализа разработаны десятки методов. Использование форматных, сигнатурных и статистических подходов варьируется по вычислительной сложности, точности, надежности, требованиям к наличию дополнительной информации [4–7]. Де-факто стандартом стало применение многомерного вектора признаков с последующей классификацией или регрессией несмотря на то, что статистический стеганоанализ позволяет делать лишь вероятностные выводы [8, 9]. Следует отметить, что соответствующая ему экспертная методика должна включать оценку достоверности полученных результатов.

Рассмотрим ситуацию, когда на экспертизу поступает графический файл $\mathbf{I}$ с вопросами:

1. Имеется в представленных файлах/файле стегановложение, выполненное с помощью программы/алгоритма <наименование программы/алгоритма>?

2. Каков размер вложения?

Допустим, в распоряжении эксперта имеется заранее обученный регрессор $R:\mathbf{X}\to\hat{\mathbf{Y}}$ где $\mathbf{X}=\left\{x_{i,j}\right\}_{i,j=1}^{i=n,j=m}$ — матрица реализаций объяс-

няющих переменных (вектора стеганоаналитических признаков), n — количество реализаций, m — количество объясняющих переменных (размер вектора признаков); $\hat{\mathbf{Y}} = \{\hat{y}_i\}_{i=1}^{j=1}$ — вектор-столбец прогнозных значений зависимой переменной.

При этом регрессор обучен на нужной стеганопрограмме/алгоритме, известны метрики качества регрессора. Также эксперту доступен стеганоаналитический алгоритм SA: $\mathbf{I} \to \mathbf{X_I}$, использованный для обучения, и он может получить вектор признаков.

Вопросы, на которые должен ответить эксперт перед началом исследования:

1. Возможно ли применение регрессора для данного файла?

2. Какова достоверность полученного результата?

3. В какой форме будет сделан вывод?

Первый вопрос относится к известной проблеме Cover-Source Mismatch (CSM), дословно — несоответствие источника контейнера. Изначально под этим понималось, что обучающие и проверяемые изображения получены разными устройствами (image acquisition), но затем понятие CSM распространилось на процессы преобразования (image processing, JPEG compression) и даже семантику изображения [10].

Проблема CSM имеет два диаметрально противоположных решения [11].

Атомистический подход. Если имеется возможность определить параметры изображения, то формируется гомогенный относительно подозрительного контейнера набор данных, т. е. происходит имитация источника подозрительного контейнера [12].

Холистический подход. Обучающее множество формируется из контейнеров, порожденных разнообразными источниками. Задача состоит в получении не столько точной модели, сколько обладающей обобщающей способностью [13].

Отдельно следует отметить адаптационный подход [14]. Идея адаптации заключается в обучении на источнике контейнеров, называемом исходным, и использовании полученных знаний для адаптации к неизвестному источнику контейнеров, называемому целевым. Этот метод позволяет детектору находить пространство, инвариантное к характеристикам, в котором распределения характеристик контейнеров исходного и целевого источников близки.

Настоящая статья посвящена реализации атомистического подхода для частного случая — стеганографии, локализованной в частотной области изображения. В широком смысле статья имеет отношение к формированию обучающего множества при машинном обучении, чему посвящены работы [15–17].

Идея статьи частично перекликается с высказанной в [18], где предложено перед финальным обнаружением добавить этап предварительной фильтрации контейнеров (отбор «хороших» контейнеров, в которых наличие/отсутствие внедренной информации может быть определено более достоверно, чем во всем множестве).

Статья соответствует направлению по формированию и проверке эффективности векторов признаков с возможностью управления соотношением точность/ресурсоемкость [19–21].

Обоснование идеи исследования

Атомистический подход предполагает формирование обучающей и тестовой выборок из одного множества. Это означает, что характеристики элементов этого множества должны лежать в определенных границах. Под характеристиками в широком смысле можно понимать характеристики трех основных сущностей, участвующих в системе формирования цифрового изображения (DIC): сцены (S), устройства (D), процесса преобразования (P):

$$DIC\colon S \times D \times P \to \mathbf{I}.$$

В узком смысле характеристики ($\mathbf{CH}$) — это производные, полученные (DER) от непосредственно изображения $\mathbf{I}$:

$$DER\colon \mathbf{I} \to \mathbf{CH_I}.$$

Значительное количество работ посвящено поиску оптимального набора характеристик изображения, с помощью которых можно относить изображения к одному множеству [22–24]. Однако данный подход имеет два недостатка:

1) реальные изображения, как правило, значительных размеров. Кумулятивные характеристики всего изображения могут не соответствовать характеристикам сегментов изображения;

2) априори неизвестно, содержит ли файл, представленный на исследование, вложение. Если содержит, то его характеристики будут искажены. Для нивелирования влияния возможной модификации файлы подвергают субдискретизации (downsampling) [25] или калибровке [26].

Идея настоящего исследования в следующем. Поскольку решающими характеристиками (функциями от изображения) при стеганоанализе являются реализации стеганоаналитических алгоритмов, то целесообразно сравнивать файлы на принадлежность к одному множеству именно через них, т. е. $\mathbf{CH_I} = \mathbf{X_I}$.

Однако именно эти характеристики наиболее чувствительны к стегановложению. В условиях априорной неизвестности наличия/размера вложения в исследуемом файле сравнение возможно только при устранении искажений, вызванных вложением. Поскольку это невозможно, остается «пожертвовать» составляющими изображения, где локализованы искажения. Для пространственных областей изображений это возможно, например, для алгоритмов семейства Least Significant Bit Replacement, где искажения затрагивают лишь плоскость младшего бита, и ее можно заполнить нулями, единицами или чередованием нулей и единиц [27]. Однако уже к LSB Matching предлагаемый подход неприменим.

Для частотной области предложено [26] использовать калибровку. «Калиброванное изображение» — изображение, очищенное от стеганографического искажения, но сохранившее семантику. «Калиброванное JPEG-изображение» получается следующим образом. Изображение разворачивается из частотного в пространственное представление, обрезается на несколько пикселей по обоим направлениям, опять сжимается в JPEG с прежними параметрами. «Калиброванное изображение» сохраняет свойства исходного на макроуровне.

Итак, предлагается решить проблему CSM путем формирования обучающего множества для анализируемого изображения из изображений с близкими значениями векторов признаков калиброванных версий.

Одним из вариантов формирования множества из близких векторов является кластеризация. Однако в практической плоскости возникает неопределенность относительно количества как кластеров, так и элементов в кластере. Исследуемый файл может попасть в кластер, мощность которого недостаточна для обучения. В этой связи кластеризация может быть использована для предварительной оценки потенциальной эффективности предложенного подхода и качества выборки.

Второй вариант — определение принадлежности изображений к одному множеству путем непосредственного расчета расстояния между векторами признаков dist($\mathbf{X_I}$, $\mathbf{X_J}$). При этом метрика должна выбираться из соображений как результативности, так и скорости вычислений.

В статье предложен подход, направленный на решение проблемы CSM для частного случая: при локализации искажений в коэффициентах дискретного косинусного преобразования (ДКП; Discrete Cosine Transform, DCT) и известном алгоритме стегановложения. Приведено формальное описание процесса анализа, архитектура, состав, технологический стек стенда, описание и результаты численного эксперимента по проверке эффективности предложенного подхода.

Формализация идеи исследования

Рассмотрим частный случай стеганоанализа. На исследование представлено/представлены изображение/изображения (SI — Suspicious Image/Images), известен стеганоалгоритм (СГА) или стеганопрограмма (СГП), с помощью которых могли быть выполнены вложения, искажения привносятся в частотную область изображения (коэффициенты дискретного косинусного преобразования JPEG). Для ответа на вопрос о размере вложения предлагается процедура (схематично изображена на рис. 1), включающая в себя следующие этапы:

1. Формирование обучающего множества (коллекции) изображений (IS — Image Set). Изображения можно получать самостоятельно фотокамерой, загружать из фотохостингов, социальных сетей и пр. В дальнейшем предполагается, что файлы изображений в необходимом количестве и формате имеются в распоряжении аналитика.

2. Формирование множества «калиброванных изображений». Калибровка как изображений IS (ClbrIS — Calibrated IS), так и изображений, переданных на исследование (ClbrSI — Calibrated SI).

3. Формирование множества пустых и заполненных изображений (EPIS — Empty and Payloaded IS). Применение СГА/СГП к коллекции изображений IS, реализация вложений с определенным шагом. Шаг определяется аналитиком исходя из требуемой точности, наличия вычислительных ресурсов и машинной памяти.

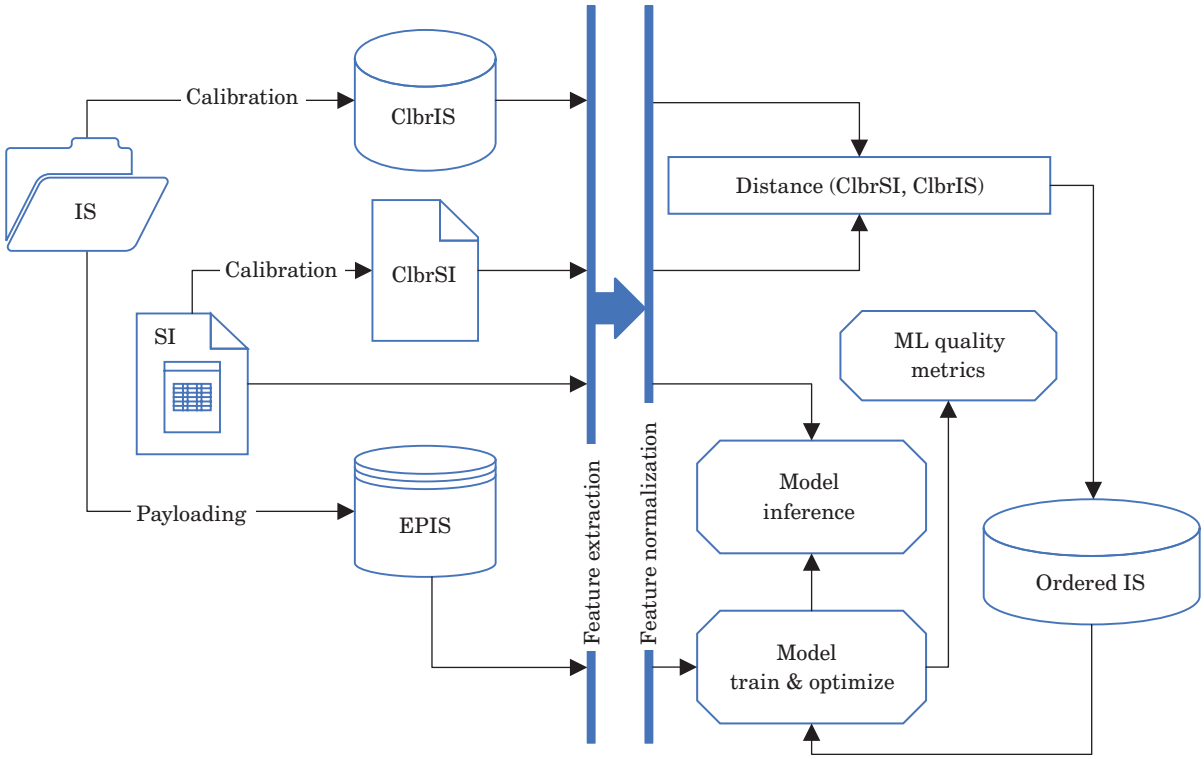
4. Формирование набора данных. Извлечение стеганоаналитических признаков (Feature extraction) из элементов EPIS и SI, нормализация (Feature normalization).

5. Формирование матрицы расстояний — Distance (ClbrSI, ClbrIS). Расчет расстояния между векторами признаков «калиброванных изображений» по выбранной метрике.

6. Упорядочивание изображений по минимуму расстояния от переданного на исследование изображения (Ordered IS). Определение размера набора данных (по мощности, по порогу расстояния, по максимуму точности распознавания).

7. Выбор модели машинного обучения и метрик качества (ML quality metrics).

8. Обучение модели на полученном наборе данных (Model train & optimize), распознавание переданного на исследование изображения (Model inference).



■ ***Рис. 1.*** Предложенная процедура стеганоанализа
■ ***Fig. 1.*** Scheme of proposed steganalytic technique

Опишем предложенную последовательность действий. Пусть:

– SI (Suspicious Image) – исследуемое изображение;

– IS (Image Set) – коллекция исходных изображений $\{IS_1, IS_2, \ldots, IS_N\}$;

– СГА/СГП – S: $S(\mathbf{I}, m, p) \to \mathbf{I}_{payloaded}$, где $\mathbf{I}$ – изображение, m – сообщение, p – параметры вложения (опционально);

– $ClbrIS = \{Calibrate(IS_i) \mid IS_i \in IS\}$ – калиброванные файлы из IS;

– $ClbrSI = Calibrate(SI)$ – калиброванное исследуемое изображение, калибровка выполняется в соответствии с [26];

– F – функция получения вектора стеганоаналитических признаков;

– $dist(\mathbf{F}_1, \mathbf{F}_2)$ – метрика расстояния между векторами;

– Q – метрика качества.

Тогда $EPIS = \{IS \cup IS_{payloaded}\}$ – множество пустых и заполненных изображений (Empty and Payloaded IS), где $IS_{payloaded} = \{S(IS_i, m_j, p_k) \mid IS_i \in IS, m_j \in M, p_k \in P\}$, M – множество стеганосообщений (файлов), P – набор параметров вложения.

Формируется набор данных относительно SI, для чего необходимо:

– вычислить векторы признаков $\mathbf{F}_{SI} = F(Clbr_{SI})$, $\mathbf{F}_i = F(ClbrIS_i)$;

– найти вектор расстояний от исследуемого изображения $\mathbf{D} = \{dist(\mathbf{F}_{SI}, \mathbf{F}_i)\}$;

– упорядочить $\mathbf{D} = (D_i)$ по возрастанию: $D\sigma_{(1)} \le D\sigma_{(2)} \le \ldots \le D\sigma_{(N)}$, где $\sigma = argsort(\mathbf{D})$ – перестановка множества индексов Ind(IS);

– выбрать подмножество из k наиболее близких к SI изображений $IS_{DS} = \{IS\sigma_{(1)}, IS\sigma_{(2)}, \ldots, IS\sigma_{(k)}\}$, где k определяется:

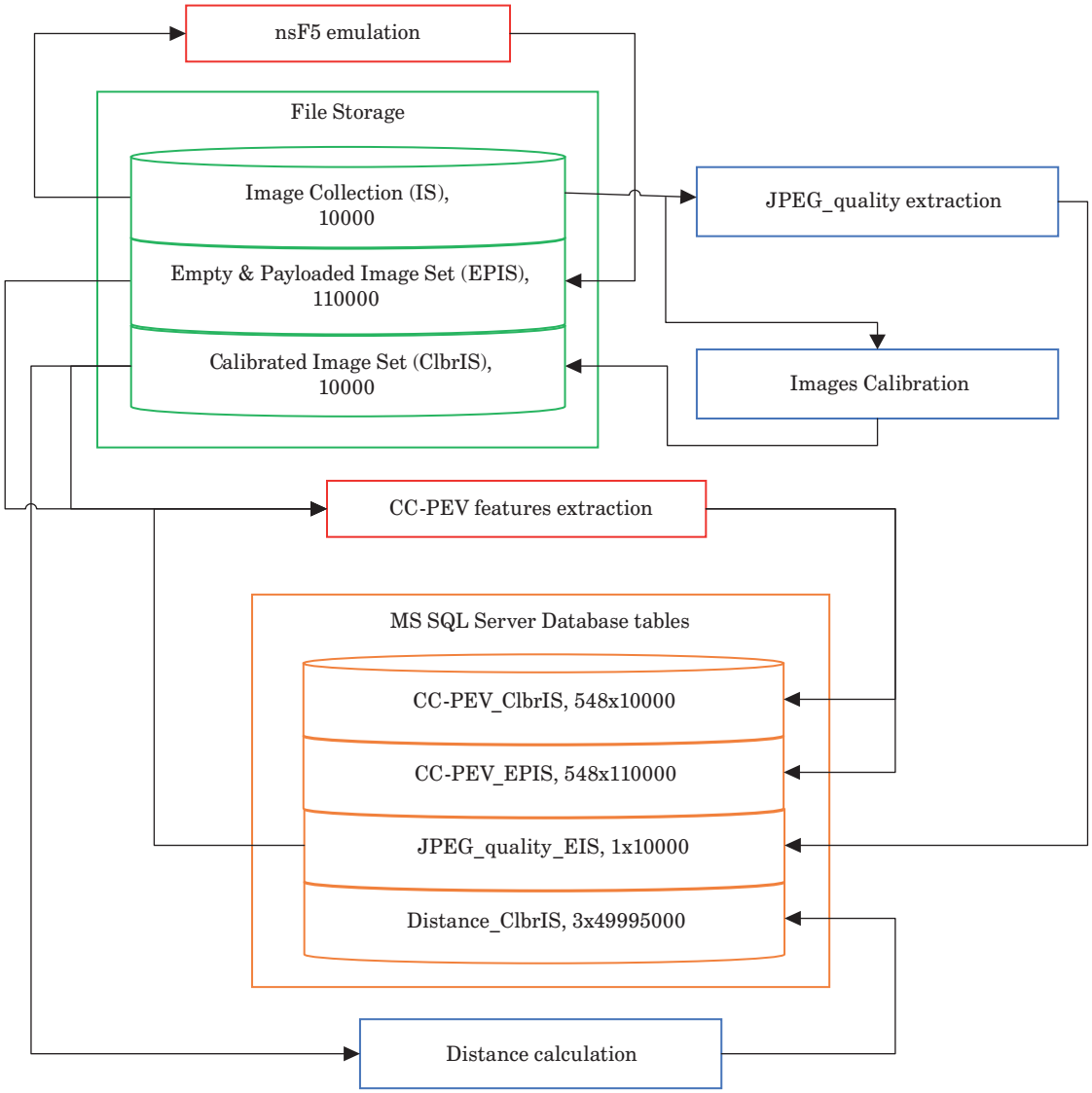
- фиксированным порогом: $k = const$;
- процентилем: $k = \lceil \alpha N \rceil$, $\alpha \in (0, 1)$;
- пороговым расстоянием между изображениями: $k = \max\{i \mid D\sigma_{(i)} \le \varepsilon\}$;
- максимумом качества распознавания: $k = \max\{i \mid Q(EPIS_{DS} \mid k = i) > Q(EPIS_{DS} \mid k = i - 1)\}$, где $EPIS_{DS} = \{EPIS_i \mid i \in Ind(IS_{DS})\}$;

– найти функцию регрессии с обучением по $EPIS_{DS}$:

$$\hat{f} = \arg\min_{f \in H} \frac{1}{|EPIS_{DS}|} \times \times \sum_{i=1}^{|EPIS_{DS}|} L\left[f(F(EPIS_{DS}^i), Pld(EPIS_{DS}^i)) \right],$$

где H – множество регрессионных функций; L – функция потерь; $Pld(EPIS_{DS}^i)$ – размер стегановложения;

– рассчитать метрики качества машинного обучения;



■ ***Рис. 2.*** Схема формирования данных для вычислительного эксперимента
■ ***Fig. 2.*** Data processing experiment scheme

– предсказать размер вложения в SI: $\mathrm{Pld}(\mathrm{SI}) = \hat{f}(F(\mathrm{SI}))$.

Экспериментальная часть

Эксперимент по определению эффективности предложенного подхода предполагает не просто вычисление метрик оценки регрессии, но и сравнение с иным подходом [28] к использованию калибровки.

Предложено [26] в состав вектора признака включать признаки как контейнера, так и его калиброванной версии. Эксперимент, проведенный на векторе признаков DCT-23 (признаки извлекаются из коэффициентов ДКП с использованием гистограмм, двумерных гистограмм, матриц совместного появления и иных функционалов), показал эффективность данного подхода.

Указанный подход получил развитие в работе [28], где сформирован вектор признаков CC-PEV-548, состоящий из набора признаков PEV-274, вычисленного по контейнеру и его калиброванной версии (Cartesian Calibration, CC): CC-PEV-548 = {CC-PEV-274 ∪ PEV-274}. В свою очередь PEV-274 [29] представляет комбинацию наборов расширенного DCT (Extended DCT-193) и Markov-81 (разности абсолютных значений коэффициентов ДКП по направлениям агрегированы в матрицы переходных вероятностей марковского процесса 1-го порядка с дальнейшим усреднением по направлениям): PEV-274 = {ExtDCT-193 ∪ Markov-81}.

В рамках эксперимента осуществляется сравнение точности определения размера стегановложения вектором признаков CC-PEV-548 по [28], а также по предложенной процедуре с PEV-274.

Цель эксперимента — проверить эффективность предложенного подхода через наличие/отсутствие эффекта от переноса части распознавательной способности CC-PEV-548 на устранение влияния CSM.

Стенд для проведения эксперимента: Intel i5-12400 2,5 GHz, SSD 500 GB, RAM 32 GB под управлением Windows 10 Pro с установленным программным обеспечением: Python 3, Visual Studio Code, MATLAB R2021, MS SQL Server 2019, MSSS Management Studio. Для данной конфигурации время на обучение по 300 контейнерам и прогноз составляет ~8 с, по 50 контейнерам ~4 с. Время на вычисление попарных расстояний 10 000 контейнеров составило ~14 сут.

Конкретизация моделей, алгоритмов, данных и инструментов, использованных в эксперименте (наборы данных и скрипты доступны в Kaggle: https://www.kaggle.com/datasets/romansolodukha/clbr-jpeg):

— коллекция изображений (IS) — первые 10 000 файлов из набора Alaska-2 (https://www.kaggle.com/c/alaska2-image-steganalysis/data): файлы JPEG с качеством (QF — Quality Factor) 95, 90, 75 в соотношении 3278/3311/3411. Для определения качества изображения использована функция get_jpg_quality (https://gist.github.com/eddy-geek/c0f01dc5401dc50a49a0a821cdc9b3e8/revisions);

— стеганоалгоритм (S) — использован nsF5 (https://dde.binghamton.edu/download/nsf5simulator), так как на нем PEV-274 показал лучшие результаты в рамках конкурса BOSS [30];

— шаг вложения — 10 % от максимально возможного (9, 19, 29…99 %), |EPIS| = 110 000;

— стеганоаналитический вектор признаков (F) — PEV-274 (https://dde.binghamton.edu/download/feature_extractors), данные нормализованы функцией MinMaxScaler из библиотеки sklearn.preprocessing;

— метрика близости (dist) — расчет осуществлен со следующими метриками из библиотеки scipy.spatial.distance: correlation (корреляционное расстояние), euclidean (евклидово расстояние), braycurtis (расстояние Брея — Кертиса), cosine (косинусное расстояние). Следует отметить, что значимых различий в результатах при использовании вышеуказанных метрик не наблюдалось. Экспериментальные данные приведены для евклидова расстояния;

— метрика качества (Q) — RMSE и коэффициент детерминации (R2). Использованы функции r2_score, mean_squared_error из библиотеки sklearn.metrics;

— регрессионная модель (f) — Ridge. Выбор регрессионной модели осуществлен на основании результатов применения AutoML на базе библиотеки lazypredict (https://lazypredict.readthedocs.io) к EPIS. Ridge оказался самым быстрым регрессором с приемлемым R2≈0,89;

— программный стек:

- MS SQL Server 2019 — база данных с таблицами данных (размерность без учета системных атрибутов приведена на рис. 2), вспомогательные представления и функции;
- MATLAB 2021 — формирование заполненных контейнеров, извлечение PEV-274, CC-PEV-548;
- Python 3.11 + Visual Studio Code — калибровка, определение качества JPEG, вычисление расстояния, формирование обучающего и тестового множеств, обучение, применение модели, вычисление метрик качества.

В эксперименте множество «калиброванных изображений» упорядочивалось по расстоянию относительно каждого файла из тестового множества. Эксперимент проведен как с учетом QF JPEG, так и без учета.

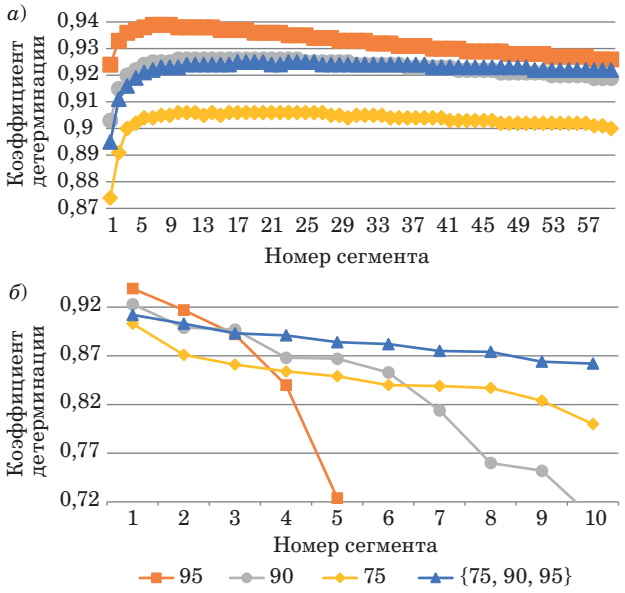
Сначала был определен минимальный размер обучающей выборки, при котором качество регрессии максимально. Для этого обучающее множество из 3000 контейнеров сегментировано по 50 контейнеров (в порядке увеличения dist — «от близких к дальним»), что с учетом вложений составляет 550 контейнеров (1-й сегмент — 50 контейнеров, наиболее близких к тестируемому файлу). Тестовая выборка — 1000 случайных исходных контейнеров, с учетом вложений — 11 000.

Относительно метрик качества следует отметить, что во всех экспериментах RMSE и R2 коррелированы с коэффициентом в диапазоне (–0,95...–0,98). В этой связи принято решение описывать результаты экспериментов только по R2. Для представления точности прогноза приведем некоторые соответствия RMSE (в процентах от максимально возможного размера стегановложения) и R2: (14; 0,7), (11,6; 0,8), (9,1; 0,9), (7,6; 0,94).

На рис. 3, *а* приведено усредненное значение R2 тестового множества в зависимости от сегмента обучающего множества с накоплением. Точка N на оси абсцисс означает, что обучение осуществлено на контейнерах, составляющих сегменты $n \le N$.

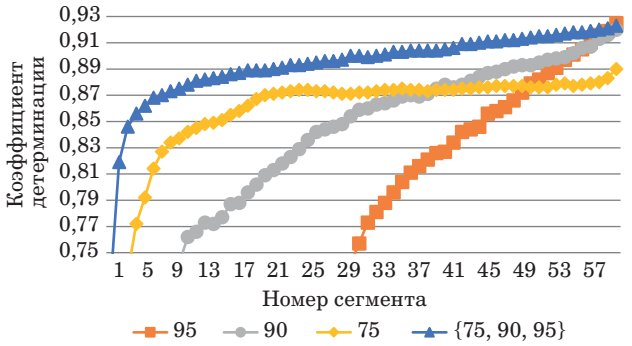
Анализ графиков рис. 3, *а* показывает, что R2 достигает максимума в зависимости от QF при $|IS_{DS}| = 300–500$, $|EPIS_{DS}| = 3300$, затем плавно уменьшается.

Дальнейшие эксперименты проводились с сегментами по 300 контейнеров (нижняя граница оптимального размера сегмента выбрана



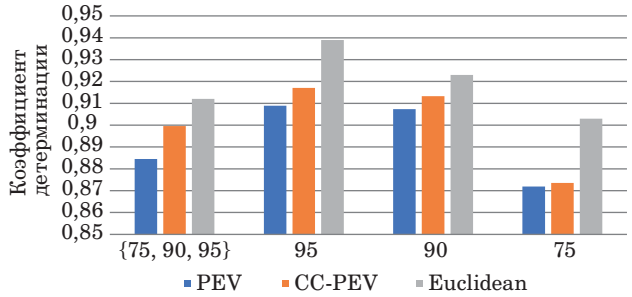
■ ***Рис. 3.*** Зависимость усредненного коэффициента детерминации от сегмента обучающей выборки с накоплением (*а*) и без накопления (*б*) (обучающее множество упорядочено по возрастанию расстояния)

■ ***Fig. 3.*** Dependence of the averaged determination coefficient on the training sample segment with accumulation (*а*) and without accumulation (*б*) (the training set is ordered by increasing distance)



■ ***Рис. 4.*** Зависимость усредненного коэффициента детерминации от сегмента обучающей выборки с накоплением (обучающее множество упорядочено по убыванию расстояния)

■ ***Fig. 4.*** Dependence of the averaged determination coefficient on the training sample segment with accumulation (the training set is ordered by decreasing distance)



■ ***Рис. 5.*** Эффективность распознавания в зависимости от подхода

■ ***Fig. 5.*** Prediction accuracy depends on the approach

из соображений минимизации времени на вычисления). На графике рис. 3, *б* приведен R2 в зависимости от сегмента (больше номер сегмента — дальше от тестового изображения). Наблюдается влияние качества изображения на скорость убывания качества распознавания. Чем выше качество JPEG, тем сильнее влияние сегментации.

Интерес вызывает поведение кривой, описывающей вариант с изображениями произвольного качества, — $Q_{\{75,90,95\}}$. Ожидалось, что эти значения должны являться усреднением значений Q_{75}, Q_{90}, Q_{95} при соответствующих сегментах (аналогично кривым рис. 3), однако, начиная с 3-го сегмента: $Q_{\{75,90,95\}} > (Q_{75} + Q_{90} + Q_{95})/3$. Это означает, что расстояние между «калиброванными изображениями» влияет на R2 сильнее, чем идентичность QF элементов обучающего и тестового множества.

Для понимания данного явления построен график рис. 4, аналогичный рис. 3, *а*, но IS_{DS} упорядочено «от дальних к близким». Кривая $Q_{\{75,90,95\}}$ находится выше Q_{75}, Q_{90}, Q_{95} вплоть до сегмента с самыми «близкими» контейнерами.

На качественном уровне это можно объяснить тем, что в сегментах, где расстояние между контейнерами мало́, идентичность QF положительно влияет на объясняющую способность модели. По мере увеличения расстояния между тестируемым контейнером и обучающим множеством отфильтрованные по QF выборки проигрывают смешанной. Это связано с тем, что на первый план выходит именно расстояние между контейнерами.

Обобщить проведенный эксперимент можно гистограммой рис. 5, где представлена сводная информация по сравнению определения размера вложения с помощью PEV-274, CC-PEV-548 и предложенного подхода в комбинации с PEV-274 (на гистограмме PEV, CC-PEV, Euclidean соответственно). Для Euclidean использован сегмент из 300 самых «близких» контейнеров. Для PEV-274, CC-PEV-548 приведено усреднение по 10 наборам данных из 300 случайных контейнеров.

Для контейнеров с QF = {90, 95} $R2_{PEV} \approx 0{,}91$, при QF = 75 наблюдается уменьшение коэффициента детерминации $R2_{PEV} \approx 0{,}87$. Использование CC-PEV-548 незначительно улучшает результаты для QF = {90, 95}, при этом для QF = 75 $R2_{PEV} \approx R2_{CC\text{-}PEV}$.

Применение предложенного подхода увеличило точность прогноза во всех группах контейнеров, что экспериментально подтвердило эффективность использования признаков, из-

влеченных из «калиброванных изображений», не в векторе признаков, а для определения расстояния между контейнерами. Другими словами, перенос информации о «калиброванном изображении» из признакового пространства в процедуру формирования релевантной обучающей выборки является более эффективным способом борьбы с CSM.

Заключение

В работе предложен, формализован и экспериментально проверен один из вариантов атомистического подхода к решению проблемы CSM в частотной области изображений. Обучающее множество формируется из контейнеров, наиболее «близких» к исследуемому файлу по расстоянию между векторами признаков их «калиброванных» версий.

Эксперимент с алгоритмом nsF5 показал, что предложенный подход обеспечивает более точную оценку размера стегановложения по сравнению с использованием как базового вектора PEV-274, так и расширенного CC-PEV-548 на случайной выборке. Установлено, что расстояние между векторами признаков по влиянию на точность прогноза преобладает над идентичностью качества JPEG элементов выборки. Предложенная процедура допускает вынесение наиболее ресурсоемкого этапа (расчет попарных расстояний) на этап предварительной подготовки, что потенциально содействует уменьшению сроков анализа.

Ограничение применения полученных результатов обусловлено частным характером валидации, так как эксперимент проведен на одном стеганоалгоритме и одном наборе данных. Обобщение результатов на другие алгоритмы и наборы изображений требует дополнительных исследований.

Таким образом, в перспективе планируется изучение границ и методики применения предложенного подхода с определением статистической достоверности результатов для каждого размера вложения. Выбор регрессионной модели обусловлен, в том числе, алгоритмической простотой и теоретической проработанностью оценки статистической значимости параметров линейной регрессии. В комбинации с оценкой достоверности приведенный в статье подход можно будет рассматривать в качестве прообраза экспертной методики.

В инфраструктурном плане для хранения векторов целесообразно осуществить миграцию с реляционной системы управления базами данных на векторную. Это связано не только с быстродействием, но и с тем, что реляционные базы данных имеют ограничения по количеству столбцов в таблицах, что приводит к невозможности нативного (файлы в строках, признаки в столбцах) представления в них многомерных стеганоаналитических векторов признаков, например Milvus или Qdrant.

Литература

1. **Cogranne R., Giboulot Q., Bas P.** ALASKA#2: Challenging academic research on steganalysis with realistic images. *IEEE International Workshop on Information Forensics and Security (WIFS)*, 2020, pp. 1–5. doi:10.1109/WIFS49906.2020.9360896
2. **Cogranne R., Giboulot Q., Bas P.** The ALASKA steganalysis challenge: A first step towards steganalysis. *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security*, 2019. doi:10.1145/3335203.3335726
3. **Bobok I. I., Kobozieva A. A.** Theoretical foundations of digital content integrity expertise. *Problemele Energeticii Regionale*, 2025, no. 1 (65), pp. 105–121. doi:10.52254/1857-0070.2025.1-65.08
4. **Солодуха Р. А., Атласов И. В., Кубасов И. А.** *Стеганоанализ цифровых изображений: технологии, алгоритмы, программная реализация*: монография. Воронеж, Воронежский институт МВД России, 2022. 172 с.
5. **Michaylov K., Sarmah D.** Steganography and steganalysis for digital image enhanced forensic analysis and recommendations. *Journal of Cyber Security Technology*, 2025, vol. 9(1), pp. 1–27. doi:10.1080/23742917.2024.2304441
6. **Сирота А. А., Дрюченко М. А., Иванков А. Ю.** Стегоанализ цифровых изображений с использованием методов поверхностного и глубокого машинного обучения: известные подходы и новые решения. *Вестник ВГУ. Серия: Системный анализ и информационные технологии*, 2021, № 1, с. 33–52. doi:10.17308/sait.2021.1/3369
7. **Полунин А. А., Яндашевская Э. А.** Использование аппарата сверточных нейронных сетей для стегоанализа цифровых изображений. *Труды ИСП РАН*, 2020, т. 32, № 4, с. 155–163. doi:10.15514/ISPRAS–2020–32(4)–11
8. **Лубин А. Ф.** О допустимости вероятностных выводов экспертного заключения в уголовном судопроизводстве. *Юридическая наука и практика: Вестник Нижегородской академии МВД России*, 2019, т. 3, № 47, с. 138–142. doi:10.36511/2078-5356-2019-3-138-142
9. **Овсянников И. В.** К вопросу о вероятном заключении эксперта. *Российская юстиция*, 2014, № 11, с. 56–59.
10. **Giboulot Q., Cogranne R., Borghys D., Bas P.** Effects and solutions of cover-source mismatch in image

steganalysis. *Signal Processing: Image Communication*, Elsevier, 2020, vol. 86. doi:10.1016/j.image.2020.115888

11. **Mallet A., Benes M., Cogranne R.** Cover-source mismatch in steganalysis: Systematic review. *EURASIP Journal on Information Security*, 2024, no. 26. doi:10.1186/s13635-024-00171-6
12. **Hou X., Zhang T., Xiong G., Wan B.** Forensics aided steganalysis of heterogeneous bitmap images with different compression history. *KSII Transactions on Internet and Information Systems*, 2012, vol. 6, no. 8, pp. 1926–1945. doi:10.3837/tiis.2012.08.003
13. **Pasquet J., Bringay S., Chaumont M.** Steganalysis with cover-source mismatch and a small learning database. *EUSIPCO: European Signal Processing Conference*, 2014, pp. 2425–2429. doi:10.5281/ZENODO.43792
14. **Abecidan R., Itier V., Boulanger J., Bas P.** Unsupervised JPEG domain adaptation for practical digital image forensics. *WIFS 2021: IEEE International Workshop on Information Forensics and Security*, 2021. doi:10.1109/WIFS53200.2021.9648397
15. **Кафтанников И. Л., Парасич А. В.** Проблемы формирования обучающей выборки в задачах машинного обучения. *Вестник Южно-Уральского государственного университета. Серия: Компьютерные технологии, управление, радиоэлектроника*, 2016, т. 16, № 3, с. 15–24. doi:10.14529/ctcr160302
16. **Парасич А. В., Парасич В. А., Парасич И. В.** Формирование обучающей выборки в задачах машинного обучения. Обзор. *Информационно-управляющие системы*, 2021, № 4, с. 61–70. doi:10.31799/1684-8853-2021-4-61-70
17. **Лебедев И. С.** Адаптивное применение моделей машинного обучения на отдельных сегментах выборки в задачах регрессии и классификации. *Информационно-управляющие системы*, 2022, № 3, с. 20–30. doi:10.31799/1684-8853-2022-3-20-30
18. **Монарев В. А., Пестунов А. И.** Повышение эффективности методов стегоанализа при помощи предварительной фильтрации контейнеров. *Прикладная дискретная математика*, 2016, № 2(32), с. 87–99. doi:10.17223/20710410/32/6
19. **Солодуха Р. А.** Стеганоанализ изображений, модифицированных алгоритмом Bit Plane Complexity Segmentation. *Информационно-управляющие системы*, 2023, № 2, с. 27–38. doi:10.31799/1684-8853-2023-2-27-38, EDN: DXURBZ
20. **Солодуха Р. А., Перминов Г. В., Атласов И. В.** Редукция набора детекторов LSB с заданной достоверностью. *Научно-технический вестник информационных технологий, механики и оптики*, 2022, т. 22, № 1, с. 74–81. doi:10.17586/2226-1494-2022-22-1-74-81
21. **Солодуха Р. А.** Повышение точности стеганоанализа пространственной области изображений за счет дополнительных стегановложений. *Информационно-управляющие системы*, 2024, № 3, с. 2–10. doi:10.31799/1684-8853-2024-3-2-10, EDN: FOOKRY
22. **Donghui Hu, Zhongjin Ma, Yuqi Fan, Lina Wang.** A study of the two-way effects of cover source mismatch and texture complexity in steganalysis. *Lecture Notes in Computer Science*, 2017, 10082, pp. 601–615. doi:10.1007/978-3-319-53465-7_45
23. **Alkhalidi M., Abu-Elnasr O., Elarif T.** A robust steganalysis method for detecting the steganography in images. *International Journal of Intelligent Computing and Information Sciences*, 2017. doi:10.21608/ijicis.2017.19819
24. **Hammad B., Ahmed I., Jamil N.** A steganalysis classification algorithm based on distinctive texture features. *Symmetry*, 2022, vol. 14, iss. 2, Art. 236. doi:10.3390/sym14020236
25. **Kato H., Osuge K., Haruta S., Sasase I.** A preprocessing by using multiple steganography for intentional image downsampling on CNN-based steganalysis. *IEEE Access*, 2020, vol. 8, pp. 195578–195593, doi:10.1109/ACCESS.2020.3033814
26. **Fridrich J.** Feature-based steganalysis for JPEG images and its implications for future design of steganographic schemes. *Information Hiding 2004. Lecture Notes in Computer Science*, 2004, vol. 3200, Springer. doi:10.1007/978-3-540-30114-1_6
27. **Solodukha R.** The CSM problem solving technique for LSBR steganography in image spatial domain. *6th International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency*, 2024, pp. 521–525. doi:10.1109/SUMMA64428.2024.10803776
28. **Kodovsky J., Fridrich J.** Calibration revisited. *Proceedings of the 11th ACM Multimedia and Security Workshop*, 2009. doi:10.1145/1597817.1597830
29. **Pevny T., Fridrich J.** Merging Markov and DCT features for multi-class JPEG steganalysis. *Proceedings SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, 2007, vol. 6505. doi:10.1117/12.696774
30. **Bas P., Filler T., Pevny T.** Break our steganographic system: The ins and outs of organizing BOSS. *Information Hiding. Lecture Notes in Computer Science*, 2011, vol. 6958. doi:10.1007/978-3-642-24178-9_5

UDC 519.6
doi:10.31799/1684-8853-2026-1-8-18
EDN: QIENHD

A novel approach to solving the CSM problem in the frequency domain of an image

R. A. Solodukha[a], PhD, Tech., Associate Professor, orcid.org/0000-0002-3878-4221, standartal@list.ru
[a]Voronezh State University of Engineering Technologies, 19, Revolucii Ave., 394036, Voronezh, Russian Federation

Introduction: One of the problems hindering the use of steganalysis in digital forensics concerns the discrepancy between the tested container and the training set. Currently available methods for solving this problem do not take into account the specifics of container distortions introduced by various steganoalgorithms. Nevertheless, a certain location of distortions allows using them to form a training set fitting the tested container. **Purpose:** To formalize and check the hypothesis on the effectiveness of training set formation based on distance ordering for "calibrated images" by feature steganalytic vectors while determining the locations of distortions in the frequency domain of images. To compare the recognition accuracy while using the proposed approach and while combining the features of a "calibrated image" and the original image into a single feature vector. **Results:** We demonstrate the feasibility of using steganalytic feature vectors of calibrated images to calculate the distance between containers. We propose the formalized procedure for the formation of a training set based on the distance between containers. We have created the software infrastructure for a numerical experiment which has shown that, regardless of the JPEG compression quality, the use of the feature vector of a calibrated image to form a training set is more effective than including it in the general feature vector. The main calculations are done to compute pairwise distances and can be carried out before the object of study appears. To ensure reproducibility of the experiment, we have presented datasets and program code in Kaggle. **Practical relevance:** Taking nsF5 steganoalgorithm as an example, we have demonstrated the advantage of applying for a random sample the PEV-274 steganoanalytical vector with the training set corresponding to the test file over CC-PEV-548. The proposed approach helps to increase the accuracy of steganalysis and to reduce the research time which is important in digital forensic practice.

Keywords — steganalysis, feature vector, nsF5, PEV-274, CC-PEV-548, steganography, Cover-Source Mismatch, machine learning, regression, distance between vectors, forensics.

For citation: Solodukha R. A. A novel approach to solving the CSM problem in the frequency domain of an image. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2026, no. 1, pp. 8–18 (In Russian). doi:10.31799/1684-8853-2026-1-8-18, EDN: QIENHD

Reference

1. Cogranne R., Giboulot Q., Bas P. ALASKA#2: Challenging academic research on steganalysis with realistic images. *IEEE International Workshop on Information Forensics and Security (WIFS)*, 2020, pp. 1–5. doi:10.1109/WIFS49906.2020.9360896
2. Cogranne R., Giboulot Q., Bas P. The ALASKA steganalysis challenge: A first step towards steganalysis. *Proceedings of the ACM Workshop on Information Hiding and Multimedia Security*, 2019. doi:10.1145/3335203.3335726
3. Bobok I. I., Kobozieva A. A. Theoretical foundations of digital content integrity expertise. *Problemele Energeticii Regionale*, 2025, no. 1 (65), pp. 105–121. doi:10.52254/1857-0070.2025.1-65.08
4. Solodukha R. A., Atlasov I. V., Kubasov I. V. *Steganoanaliz cifrovyh izobrazhenij: tekhnologii, algoritmy, programmnaya realizaciya* [Steganalysis of digital images: technologies, algorithms, software implementation]. Voronezh,Voronezhskij institut Ministerstva vnutrennih del Rossii Publ., 2022. 172 p. (In Russian).
5. Michaylov K., Sarmah D. Steganography and steganalysis for digital image enhanced forensic analysis and recommendations. *Journal of Cyber Security Technology*, 2025, vol. 9(1), pp. 1–27. doi:10.1080/23742917.2024.2304441
6. Sirota A. A., Dryuchenko M. A., Ivankov A. Y. Steganalysis of digital images by means of shallow and deep machine learning: existing approaches and new solutions. *Proceedings of Voronezh State University. Series: Systems Analysis and Information Technologies*, 2021, no. 1, pp. 33–52 (In Russian). doi:10.17308/sait.2021.1/3369
7. Polunin A. A., Yandashevskaya E. A. Using of convolutional neural networks for steganalysis of digital images. *Proceedings of the Institute for System Programming of the RAS*, 2020, vol. 32, iss. 4, pp. 155–164 (In Russian). doi:10.15514/ISPRAS–2020–32(4)–11
8. Lubin A. F. About admission of probabilistic conclusions of expert conclusion in a criminal trial. *Legal Science and Practice: Journal of Nizhny Novgorod Academy of the Ministry of Internal Affairs of Russia*, 2019, vol. 3, no. 47, pp. 138–142 (In Russian). doi:10.36511/2078-5356-2019-3-138-142
9. Ovsyannikov I. V. To question about probabilistic conclusion of expert. *Rossijskaya yusticiya*, 2014, no. 11, pp. 56–59 (In Russian).
10. Giboulot Q., Cogranne R., Borghys D., Bas P. Effects and solutions of cover-source mismatch in image steganalysis. *Signal Processing: Image Communication*, Elsevier, 2020, vol. 86. doi:10.1016/j.image.2020.115888
11. Mallet A., Benes M., Cogranne R. Cover-source mismatch in steganalysis: Systematic review. *EURASIP Journal on Information Security*, 2024, no. 26. doi:10.1186/s13635-024-00171-6
12. Hou X., Zhang T., Xiong G., Wan B. Forensics aided steganalysis of heterogeneous bitmap images with different compression history. *KSII Transactions on Internet and Information Systems*, 2012, vol. 6, no. 8, pp. 1926–1945. doi:10.3837/tiis.2012.08.003
13. Pasquet J., Bringay S., Chaumont M. Steganalysis with cover-source mismatch and a small learning database. *EUSIPCO: European Signal Processing Conference*, 2014, pp. 2425–2429. doi:10.5281/ZENODO.43792
14. Abecidan R., Itier V., Boulanger J., Bas P. Unsupervised JPEG domain adaptation for practical digital image forensics. *WIFS 2021: IEEE International Workshop on Information Forensics and Security*, 2021. doi:10.1109/WIFS53200.2021.9648397
15. Kaftannikov I. L., Parasich A. V. Problems of training set's formation in machine learning tasks. *Bulletin of the South Ural State University. Ser. Computer Technologies, Automatic Control, Radio Electronics*, 2016, vol. 16, no. 3, pp. 15–24. (In Russian). doi:10.14529/ctcr160302
16. Parasich A. V., Parasich V. A., Parasich I. V. Training set formation in machine learning tasks. Survey. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2021, no. 4, pp. 61–70 (In Russian). doi:10.31799/1684-8853-2021-4-61-70
17. Lebedev I. S. Adaptive application of machine learning models on separate segments of a data sample in regression and classification problems. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2022, no. 3, pp. 20–30 (In Russian). doi:10.31799/1684-8853-2022-3-20-30
18. Monarev V. A., Pestunov A. I. Enhancing steganalysis accuracy via tentative filtering of stego-containers. *Applied Discrete Mathematics*, 2016, no. 2 (32), pp. 87–99 (In Russian). doi:10.17223/20710410/32/6
19. Solodukha R. Steganalysis of Bit Plane Complexity Segmentation algorithm. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2023, no. 2, pp. 27–38 (In Russian). doi:10.31799/1684-8853-2023-2-27-38, EDN: DXURBZ
20. Solodukha R. A., Perminov G. V., Atlasov I. V. Reduction of LSB detectors set with definite reliability. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2022, vol. 22, no. 1, pp. 74–81 (In Russian). doi:10.17586/2226-1494-2022-22-1-74-81
21. Solodukha R. A. Increasing the accuracy of spatial domain steganalysis through additional embeddings. *Informatsionno-upravliaiushchie sistemy* [Information and Control Sys-

tems], 2024, no. 3, pp. 2–10 (In Russian). doi:10.31799/1684-8853-2024-3-2-10, EDN: FOOKRY
22. Donghui Hu, Zhongjin Ma, Yuqi Fan, Lina Wang. A study of the two-way effects of cover source mismatch and texture complexity in steganalysis. *Lecture Notes in Computer Science*, 2017, 10082, pp. 601–615. doi:10.1007/978-3-319-53465-7_45
23. Alkhalidi M., Abu-Elnasr O., Elarif T. A robust steganalysis method for detecting the steganography in images. *International Journal of Intelligent Computing and Information Sciences*, 2017. doi:10.21608/ijicis.2017.19819
24. Hammad B., Ahmed I., Jamil N. A steganalysis classification algorithm based on distinctive texture features. *Symmetry*, 2022, vol. 14, iss. 2, Art. 236. doi:10.3390/sym14020236
25. Kato H., Osuge K., Haruta S., Sasase I. A preprocessing by using multiple steganography for intentional image downsampling on CNN-based steganalysis. *IEEE Access*, 2020, vol. 8, pp. 195578–195593, doi:10.1109/ACCESS.2020.3033814
26. Fridrich J. Feature-based steganalysis for JPEG images and its implications for future design of steganographic schemes. *Information Hiding 2004. Lecture Notes in Computer Science*, 2004, vol. 3200, Springer. doi:10.1007/978-3-540-30114-1_6
27. Solodukha R. The CSM problem solving technique for LSBR steganography in image spatial domain. *6th International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency*, 2024, pp. 521–525. doi:10.1109/SUMMA64428.2024.10803776
28. Kodovsky J., Fridrich J. Calibration revisited. *Proceedings of the 11th ACM Multimedia and Security Workshop*, 2009. doi:10.1145/1597817.1597830
29. Pevny T., Fridrich J. Merging Markov and DCT features for multi-class JPEG steganalysis. *Proceedings SPIE, Electronic Imaging, Security, Steganography, and Watermarking of Multimedia Contents IX*, 2007, vol. 6505. doi:10.1117/12.696774
30. Bas P., Filler T., Pevny T. Break our steganographic system: The ins and outs of organizing BOSS. *Information Hiding. Lecture Notes in Computer Science*, 2011, vol. 6958. doi:10.1007/978-3-642-24178-9_5

УВАЖАЕМЫЕ АВТОРЫ!

Научные базы данных, включая Scopus и Web of Science, обрабатывают данные автоматически. С одной стороны, это ускоряет процесс обработки данных, с другой — различия в транслитерации ФИО, неточные данные о месте работы, области научного знания и т. д. приводят к тому, что в базах оказывается несколько авторских страниц для одного и того же человека. В результате для всех по отдельности считаются индексы цитирования, что снижает рейтинг ученого.

Для идентификации авторов в сетях Thomson Reuters проводит регистрацию с присвоением уникального индекса (ID) для каждого из авторов научных публикаций.

Процедура получения ID бесплатна и очень проста, есть возможность провести регистрацию на 12 языках, включая русский (чтобы выбрать язык, кликните на зеленое поле вверху справа на стартовой странице): https://orcid.org

УДК 004.934.2
doi:10.31799/1684-8853-2026-1-19-35
EDN: DSRKFE

Обзоры
Surveys

Аналитический обзор применения больших языковых моделей для автоматического распознавания речи

И. С. Кипяткова[a], *канд. техн. наук, доцент, старший научный сотрудник, orcid.org/0000-0002-1264-4458, kipyatkova@iias.spb.su*
М. Д. Долгушин[a], *младший научный сотрудник, orcid.org/0000-0002-4344-2330*
И. А. Кагиров[a], *научный сотрудник, orcid.org/0000-0003-1196-1117*
[a]*Санкт-Петербургский Федеральный исследовательский центр РАН, 14-я линия В. О., 39, Санкт-Петербург, 199178, РФ*

Введение: *одной из тенденций в области обработки естественных языков является использование больших языковых моделей. В системах распознавания речи большие языковые модели начинают заменять традиционные благодаря их способности учитывать более широкий контекст.* ***Цель:*** *выполнить систематизацию и обобщение существующих методов совместного использования систем автоматического распознавания речи и больших языковых моделей.* ***Результаты:*** *выявлены основные тенденции внедрения больших языковых моделей в процесс распознавания речи. Анализ продемонстрировал, что применение больших языковых моделей для переоценки гипотез и коррекции ошибок распознавания стабильно улучшает результаты распознавания, хотя это улучшение не всегда является принципиальным и сопряжено с риском генерации недостоверной информации вследствие возможных галлюцинаций моделей. Установлено, что контекстуализация и контекстное обучение больших языковых моделей могут как значительно улучшать, так и, в некоторых случаях, ухудшать результаты распознавания.* ***Практическая значимость:*** *полученные выводы могут найти практическое применение при создании систем автоматического распознавания речи на различных естественных и малоресурсных языках, а также для речи с переключением кодов.* ***Обсуждение:*** *установлено, что рекуррентные и диффузионные архитектуры больших языковых моделей пока не получили широкого распространения в задачах распознавания речи, однако обладают значительным потенциалом. Отмечена тенденция к использованию декодерных архитектур, что в свою очередь порождает проблемы галлюцинаций и ориентации на письменные нормы при генерации текста.*

Ключевые слова *— большие языковые модели, переоценка гипотез, коррекция ошибок, контекстное обучение, автоматическое распознавание речи.*

Для цитирования: Кипяткова И. С., Долгушин М. Д., Кагиров И. А. Аналитический обзор применения больших языковых моделей для автоматического распознавания речи. *Информационно-управляющие системы*, 2026, № 1, с. 19–35. doi:10.31799/1684-8853-2026-1-19-35, EDN: DSRKFE
For citation: Kipyatkova I. S., Dolgushin M. D., Kagirov I. A. Analytical review of the application of large language models for automatic speech recognition. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2026, no. 1, pp. 19–35 (In Russian). doi:10.31799/1684-8853-2026-1-19-35, EDN: DSRKFE

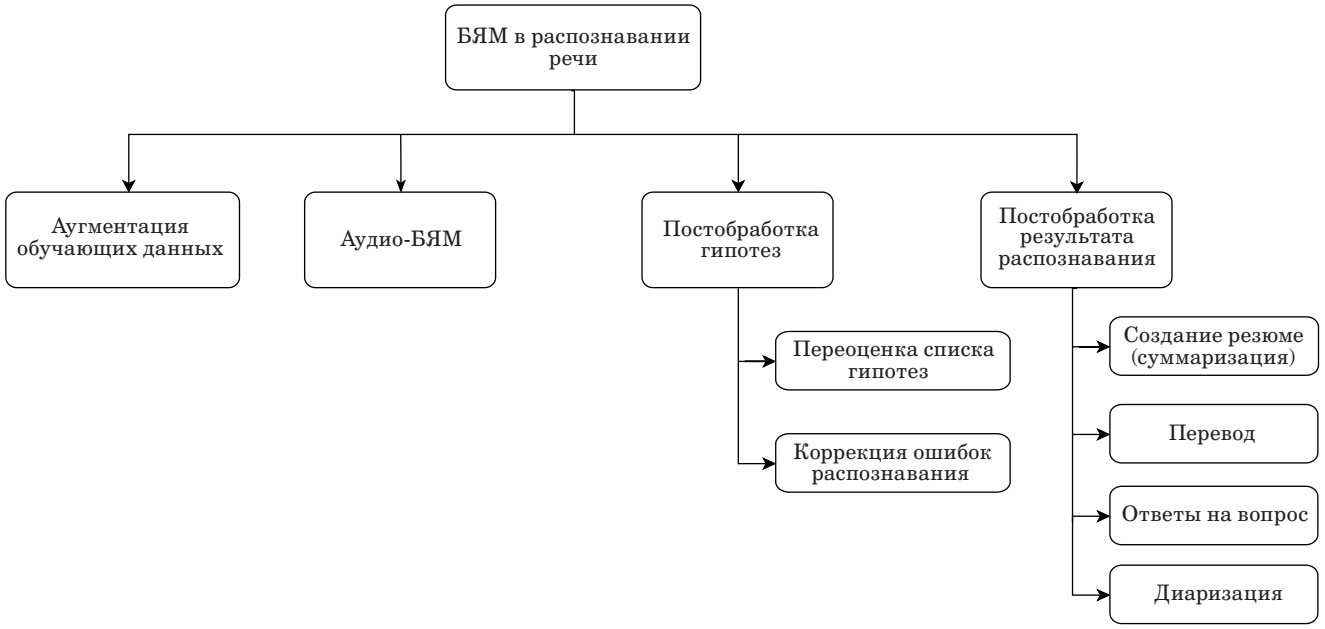
Введение

Большие языковые модели (БЯМ; large language models, LLM) в настоящее время все чаще применяются для решения широкого круга задач из области обработки естественных языков: от обнаружения ошибок в документах до ответов на вопросы по содержанию текста и машинного перевода. При этом БЯМ успешно работают не только с текстовой модальностью, но еще и с аудио- и видеоданными.

Целью настоящего обзора является систематизация и обобщение существующих методов применения БЯМ для распознавания речи. Стратегии применения БЯМ достаточно разнообразны: БЯМ могут использоваться как отдельный модуль для исправления ошибок, допущенных основной системой распознавания речи (СРР), как модуль для переоценки и выбора лучшей гипотезы распознавания или же быть интегрированы в единую архитектуру вместе с речевым кодером (рис. 1). Помимо этого, способности БЯМ к генерации текста оказываются полезными при аугментации обучающих данных. Наконец, БЯМ могут использоваться для различных задач, относящихся к постобработке результатов распознавания, например для генерации ответов в диалоговых системах или для машинного перевода.

Основные архитектуры БЯМ

Большая языковая модель — предобученная языковая модель, которая состоит из нейронной сети со множеством параметров (10 млрд и более), обученной на большом количестве неразмеченного текста или иных данных [1, 2]. В некоторых работах [3] предобученные языковые модели, имеющие менее 10 млрд параметров, но



■ ***Рис. 1.*** Основные области применения БЯМ
■ ***Fig. 1.*** Main application areas of LLMs

демонстрирующие показатели, сопоставимые с таковыми у БЯМ, называются малыми языковыми моделями (small language models).

В большинстве фундаментальных языковых моделей (foundation models) применяется архитектура трансформер [4, 5]. Базовая архитектура трансформер подразумевает два основных блока: кодер, преобразующий входную последовательность в скрытое представление, и декодер, порождающий выходную последовательность с использованием как скрытого представления, так и предыдущих элементов выходной последовательности. Архитектуры современных БЯМ можно разбить на три класса: архитектуры типа «кодер-декодер», «только кодер» и «только декодер». С 2024 г. появились БЯМ, в которых применяются альтернативные архитектуры, например модели непрерывного пространства состояний [6] и диффузионные модели [7].

Архитектура «кодер-декодер» (классическая архитектура модели трансформер) используется для так называемых задач преобразования последовательности в последовательность (sequence-to-sequence), актуальных, в том числе, в рамках машинного перевода и распознавания речи. Подобную архитектуру имеют предобученные модели BART [8], mBART [9], T5 [10] и mT5 [11].

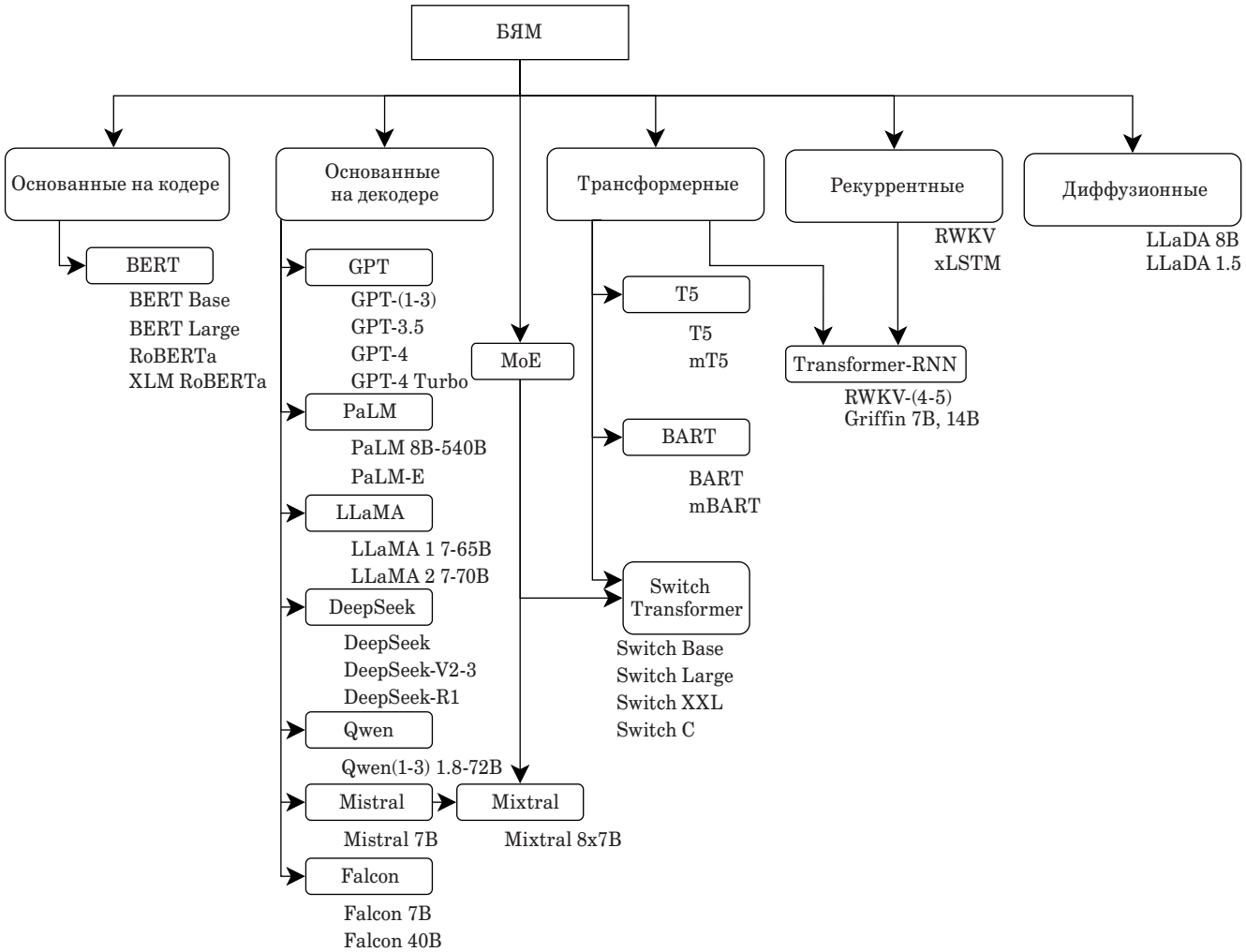
В моделях, состоящих только из кодера, каждый слой содержит механизм самовнимания и полносвязную сеть. Самовнимание является двунаправленным, т. е. при обработке токена модель может учитывать контекст как слева (предшествующие токены), так и справа (последующие токены). Эта архитектура применяется в задачах, связанных с векторными представлениями входной последовательности и не требующих генерации новых последовательностей, например при классификации текста, распознавании именованных сущностей, анализе тональности текста. Одной из моделей такого типа является модель BERT [12], которая в работе [13] была адаптирована для работы с русским языком (ruBERT). В работе [14] представлена RoBERTa, улучшенная модель BERT с особым процессом предобучения, обладающая повышенной устойчивостью к шумам. В работе [15] данная модель была адаптирована для задачи многоязычной обработки.

В моделях, состоящих только из декодера, самовнимание однонаправленное, т. е. при обработке токена модель обращает внимание только на текущий и предшествующий токены. Принцип работы такой модели авторегрессионный: она обрабатывает входную последовательность и на основе этого анализа генерирует следующий токен. Затем этот сгенерированный токен добавляется к последовательности, и процедура повторяется. Такая модель используется, прежде всего, для задач генерации текста. В качестве примера можно привести GPT [16], PaLM [17], LLaMA [18], Falcon [19] и Mistral [20].

Поскольку с ростом числа параметров модели увеличиваются как объем вычислений, так и требования к ресурсам, разработка методов оптимизации является актуальной задачей. Так,

к «оптимизированным» архитектурам можно отнести «смесь экспертов» (Mixture of Experts, MoE), принцип работы которой состоит в том, что вся модель делится на некоторое количество «экспертов» — отдельных подмоделей в общей архитектуре, которые в процессе обучения раздельно обрабатывают разные аспекты входных данных. При этом для отбора наиболее подходящих результатов используется дополнительная модель меньшего размера (распределитель). Дальнейшее развитие этой архитектуры привело к появлению «разреженных смесей экспертов» (sparse MoE), в которых распределитель не просто обрабатывает ответы, а предварительно выбирает наиболее подходящих экспертов для конкретной задачи, отключая остальные, что значительно снижает вычислительную нагрузку. Примером такого подхода служит модель Mixtral [21], являющаяся развитием декодерной модели Mistral. Mixtral использует архитектуру разреженной смеси экспертов, при которой для решения каждой задачи одновременно активируются только два из восьми экспертов. Другим примером является работа [22], в которой смесь экспертов совместно с архитектурой T5 использовалась для оптимизации трансформера Switch Transformer, что в итоге привело к значительному увеличению числа параметров модели, увеличению скорости обучения и снижению ресурсозатратности [22].

Рекуррентные БЯМ представляют собой класс архитектур, которые либо полностью основаны на рекуррентных нейронных сетях, либо сочетают в себе рекуррентные и трансформерные подходы. Известно, что трансформерные архитектуры легко распараллеливаются и допускают значительное увеличение объема обучаемых параметров, однако потребление памяти и вычислительная сложность растут квадратично относительно длины входной последовательности. В отличие от трансформеров, рекуррентные нейронные сети демонстрируют линейную зави-



■ ***Рис. 2.*** Основные архитектуры БЯМ
■ ***Fig. 2.*** Principal architectures of LLMs

симость затрат памяти и вычислительной сложности от длины последовательности, однако их параллелизация и масштабирование затруднены. Это послужило толчком к созданию новых архитектур, таких как Mamba [23], основанная на модели пространства состояний, или гибридных рекуррентно-трансформерных БЯМ, например RWKV [24], xLSTM [25] или Griffin [26].

Диффузионные БЯМ [27] представляют собой альтернативный подход к созданию генеративных нейросетевых моделей для восстановления маскированного и зашумленного текста. Они позволяют порождать текст без использования авторегрессии [28], а также моделировать двунаправленные зависимости между токенами. Языковые модели, основанные на диффузионных моделях, показывают превосходные результаты в задачах, связанных с выводом обратных заключений из заданных утверждений, однако исследования по их применению в области речевых технологий пока что не успели получить широкого распространения [29].

Представленные архитектуры БЯМ (рис. 2) иллюстрируют факт частого объединения различных архитектур в гибридные для уменьшения недостатков каждой из них. Также из схемы следует, что диффузионные языковые модели пока что получили меньшее распространение по сравнению с трансформерными и рекуррентными. Однако представляется, что высокая устойчивость диффузионных моделей к шумам обладает потенциалом в контексте задач малоресурсного языкового моделирования, когда обучающие данные невелики. В настоящее время БЯМ находят все большее применение для различных задач, в том числе для автоматического распознавания речи. В последующих разделах приведен обзор основных методов применения БЯМ для распознавания речи.

Применение БЯМ для переоценки списка гипотез

Для повышения точности распознавания часто используется метод переранжирования гипотез. Этот процесс начинается с того, что на первом этапе СРР генерирует не одну окончательную версию, а несколько наиболее вероятных вариантов или гипотез. Из них формируется список N лучших гипотез (N указывает на количество предложенных системой вариантов с наивысшими оценками). БЯМ вычисляет новые оценки для каждой гипотезы. Далее, на втором этапе, эти предварительно отобранные гипотезы подвергаются дополнительной оценке: БЯМ вычисляет новые, уточненные оценки для каждой гипотезы. В итоге исходная вероятностная оценка от СРР объединяется с оценкой, полученной от БЯМ, следующим образом:

$$w_{best} = \text{argmax}_{w \in W}[(1 - \lambda)\log P_{\text{СРР}}(w) + \lambda \log P_{\text{БЯМ}}(w)],$$

где w_{best} — выходная гипотеза с наибольшей вероятностью; w — гипотеза из списка лучших гипотез; λ — весовой коэффициент БЯМ; $P_{\text{СРР}}$, $P_{\text{БЯМ}}$ — вероятности гипотезы, полученные от СРР и БЯМ.

После этого выполняется переранжирование гипотез распознавания в соответствии с новыми вероятностными оценками и осуществляется выбор новой наилучшей гипотезы, т. е. гипотезы с наибольшей вероятностью. Аналогичным образом может выполняться переоценка не списка гипотез, а решетки слов, которая представляет собой граф гипотез с их вероятностными оценками.

Использование моделей языка на основе архитектуры BERT для переоценки гипотез распознавания подробно рассмотрено в работе [30]. Эксперименты на корпусе LibriSpeech показали, что применение BERT для повторной оценки 100 наиболее вероятных гипотез значительно повышает качество распознавания по сравнению с однонаправленными моделями.

В исследовании [31] задача переранжирования гипотез сформулирована как предсказание гипотезы с минимальным значением показателя неправильно распознанных слов (word error rate, WER) из списка N лучших гипотез. Именно эту идею авторы использовали при создании модели.

Работа [32] предлагает подход к переранжированию гипотез с использованием многомодальных БЯМ, объединяющих текстовые и речевые токены. Для получения речевых представлений используется HuBERT.

Важно отметить, что ограниченность списков лучших гипотез приводит к потере альтернативных вариантов. Именно поэтому в работе [33] предлагается переоценка всей решетки распознавания, что позволяет хранить больше гипотез в графовой структуре. БЯМ получает полное пространство гипотез и выдает единственную наилучшую. Сравнение показало эффективность метода на коротких фразах, но обнаружило ухудшение результатов для длинных фраз как при переоценке гипотез, так и при переоценке решетки. Авторы работы предполагают, что это связано с малыми размерами модели, а также с особенностью обучающих данных, представляющих собой небольшой набор спонтанной японской речи, поэтому длинных фраз в них немного. Также, предположительно, это может быть связано с ограничением длины контекста. Список

лучших гипотез или решетка для длинного предложения могут превысить длину контекста, которую способна обработать БЯМ. Кроме того, с увеличением длины фразы увеличивается возможное число ошибок, а ошибка в одном слове может приводить к ошибкам в других словах.

Применение БЯМ для коррекции ошибок

Большие языковые модели доказали свою высокую эффективность в исправлении орфографических и грамматических ошибок, что сделало их эффективным инструментом для корректировки гипотез, полученных в результате распознавания речи. В этой области выделяют два основных подхода: неограниченную и ограниченную коррекцию ошибок [34]. При неограниченной коррекции БЯМ генерирует полностью новую, исправленную гипотезу, опираясь на входной список лучших вариантов. Однако такой подход может приводить к избыточным изменениям, особенно когда обрабатывается лишь одна гипотеза.

В противоположность этому ограниченная коррекция ошибок требует от БЯМ выбора одной из предложенных гипотез, не допуская создания принципиально новых. Этот подход реализуется двумя способами: либо через селективный метод, при котором БЯМ непосредственно выбирает гипотезу из заданного списка, что по сути аналогично переоценке списка гипотез, либо через метод ближайшего соответствия, когда БЯМ сначала генерирует скорректированную гипотезу, а затем выбирает из исходного списка ту, что максимально близка (по показателю расстояние Левенштейна) к сгенерированному исправлению. Основное преимущество ограниченной коррекции заключается в снижении риска внесения новых ошибок, так как конечный результат всегда находится в пределах гипотез, изначально полученных от СРР. Тем не менее качество такой коррекции напрямую зависит от разнообразия и представительности этих исходных гипотез.

Различные БЯМ и стратегии их применения активно используются для оптимизации механизма коррекции ошибок. Так, в [35] подчеркивается важность стратегий расширения данных для обучения робастных моделей, а в [36] указывается эффективность метода ограничения коррекции решеткой распознавания. Современные крупные БЯМ, такие как ChatGPT, также активно применяются в задачах коррекции ошибок. Исследования [34, 37] показывают, что, хотя эти модели могут давать сопоставимые или лучшие результаты для определенных архитектур распознавания (например, Transformer-Transducer), их эффективность снижается для моделей, которые осуществляют сильную нормализацию текста, например Whisper. Это связано с тем, что нормализация уменьшает разнообразие гипотез, ограничивая пространство для коррекции. Одной из альтернатив решения этой проблемы является комбинация списков N лучших гипотез от различных систем распознавания, что позволяет использовать ошибки различных типов и улучшить общий результат коррекции [37].

С точки зрения вычислительных ресурсов рациональным является возможность коррекции только одной наилучшей гипотезы, как это предложено в [38] с применением многоязычной БЯМ Qwen1.5 7B. Этот подход позволил авторам снизить ресурсоемкость и заодно продемонстрировал эффективность переноса знаний между языками со схожей письменностью.

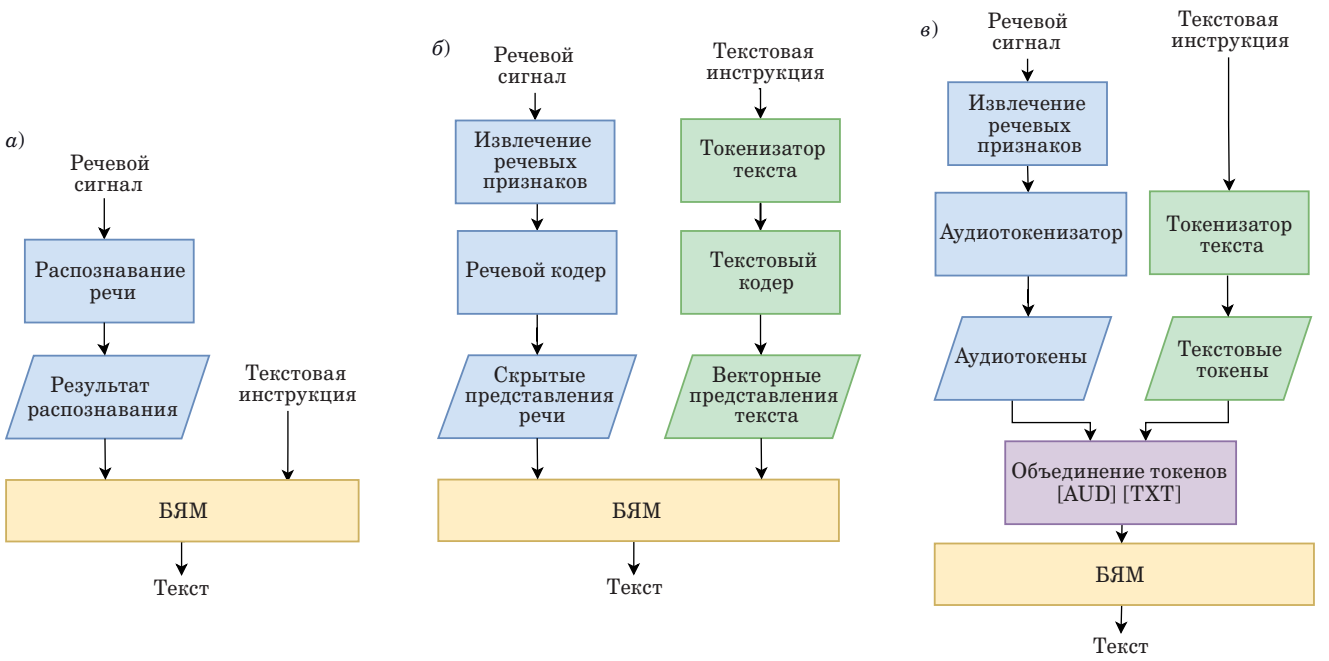
Серьезной проблемой при использовании БЯМ остается риск так называемых «галлюцинаций» — генерации недостоверной информации моделью, особенно если исходные ошибки системы распознавания минимальны. В таких случаях БЯМ может вносить ошибочные изменения, что в итоге приводит к увеличению значения показателя WER [39]. Для решения этой проблемы в [39] предлагается многопроходный метод коррекции, объединяющий результаты от различных систем распознавания и разных БЯМ с использованием алгоритма ROVER и компенсирующий недостатки отдельных моделей, что приводит к общему снижению риска галлюцинаций.

В целом использование БЯМ для коррекции ошибок часто дает лучшие результаты и происходит быстрее, чем переранжирование. Однако необходимо учитывать, что БЯМ могут генерировать синонимы, отличающиеся по звучанию от произнесенного слова [36], или вносить избыточные перефразирования [40], стремясь сделать текст более естественным. Контроль над этими изменениями может быть достигнут путем тщательного подбора запросов (prompt), как, например, сделано в [41] при обработке векторных представлений модели BART.

Объединение аудио и БЯМ

Аудио-БЯМ — это тип БЯМ, способный работать с аудиомодальностью за счет интеграции аудиоинформации в архитектуру БЯМ [42]. Основными способами интеграции аудиоинформации в БЯМ являются каскадная интеграция, интеграция на основе скрытых представлений и интеграция на основе аудиотокенов (рис. 3, *а–в*) [43].

Каскадная интеграция. Каскадная интеграция является самым простым методом ин-



■ ***Рис. 3.*** Схемы основных способов интеграции речевых и текстовых данных в аудио-БЯМ: *а* — каскадный; *б* — на основе скрытых представлений; *в* — на основе аудиотокенов

■ ***Fig. 3.*** Main approaches for integrating speech and text data into audio-LLMs: *а* — cascade; *б* — based on latent representations; *в* — based on audio tokens

теграции речевых данных в БЯМ. Речь вначале преобразуется в текст с помощью модуля автоматического распознавания речи, а затем полученный текст обрабатывается с помощью БЯМ [43]. Примером является модель, описанная в работе [44]: авторы использовали ряд фундаментальных моделей для обработки аудиоданных (в том числе Whisper для распознавания речи), при этом в качестве БЯМ-интерфейса служил ChatGPT. Преимуществом каскадного подхода является простота в реализации: во-первых, он позволяет использовать уже существующие, предварительно обученные СРР и БЯМ без необходимости дообучения и, во-вторых, не требует больших вычислительных ресурсов благодаря независимости этапов обработки данных.

Каскадная интеграция также использовалась в работе [29], в которой исследовалось применение диффузионной модели LLaDA 8B Instruct в нескольких сценариях: для улучшения результатов распознавания, полученных с помощью Whisper-LLaMA, для каскадного объединения с Whisper и в качестве самостоятельного декодера, заменяющего декодер Whisper. При каскадном объединении на вход модели подавалась не только текстовая транскрипция, но и векторные представления речевого сигнала, что привело к снижению WER на 12,3 % и улучшению результатов по сравнению с использованием Whisper-LLaMa. При этом введение только текста в качестве входных данных не позволило повысить точность распознавания. Также значительное повышение производительности продемонстрировало привлечение диффузионной модели в качестве декодера, однако данный подход не позволил достичь снижения WER.

Однако, несмотря на эти преимущества, каскадная интеграция имеет существенный недостаток: ошибки, допущенные системой СРР, неизбежно распространяются и влияют на последующую работу БЯМ. Кроме того, при таком подходе БЯМ не имеет прямого доступа к аудиопризнакам, что ограничивает ее возможности по учету нюансов речи.

Интеграция на основе скрытых представлений. Этот подход предполагает использование речевого кодера, который обрабатывает речевой сигнал и генерирует скрытые представления. Эти представления затем напрямую подаются в БЯМ, минуя промежуточный этап преобразования в текст [43]. Речевой кодер может быть обучен с нуля, или же в нем используются предварительно обученные модели, такие как HuBERT [45]. Основная сложность здесь заключается в согласовании длины последовательностей, поскольку речевые признаки обычно значительно длиннее текстовых токенов.

Первые эксперименты по такому объединению речевых и текстовых данных на основе аудиотокенов предложены в исследовании [46], где в базовую СРР была интегрирована модель BERT для работы с путунхуа. Акустические при-

знаки выравнивались с текстовыми токенами, при этом каждому слову в тексте соответствовал сегмент аудиофреймов. Далее аудиофреймы преобразовывались аудиокодером в векторные представления той же размерности, что и в BERT. Тем не менее авторам не удалось превзойти результаты базовой модели DNN-HMM.

Более поздние работы демонстрируют значительный прогресс. Например, в [47] представлена мультизадачная многоязычная модель SLM, которая обрабатывает как текстовые, так и речевые данные. Она основана на предобученной универсальной речевой модели Google USM [48] и различных моделях T5. В этой работе веса исходных моделей были заморожены, и был обучен небольшой адаптер (1 % параметров), преобразующий речевые векторные представления (эмбеддинги) в текстовые. Затем эти представления подаются в БЯМ вместе с внешними текстовыми представлениями. В работе отмечается, что, несмотря на улучшение точности работы, модель иногда может порождать галлюцинации. В статье [49] также был предложен специализированный конвертер, согласующий аудиокодер с БЯМ, что позволяет преобразовывать речевые представления в совместимое векторное пространство. Эта модель, обученная по стратегии «обучаемый аудиокодер + обучаемый конвертер + фиксированная БЯМ», показала высокие результаты во многих задачах, включая автоматическое распознавание речи и ведение диалога. Интеграция на основе скрытых представлений позволяет снизить уровень ошибки, поскольку БЯМ учитывает аудиопризнаки, которые были бы утрачены при использовании промежуточного преобразования аудио в текст. Тем не менее многие из этих подходов ориентированы на дообучение адаптера под конкретную задачу, и они обычно имеют более высокие требования к вычислительным ресурсам и объему данных для обучения.

В работе нашего коллектива [50] рассматривалось применение модели W2V2-BERT v2, основанной на объединении аудиокодера и языковой модели BERT, для малоресурсного распознавания речи на карельском языке. Продемонстрированные результаты значительно превзошли аналоги на основе дообучения только аудиокодера, аудиокодера со статистической языковой моделью и Whisper. Авторы еще одного исследования, посвященного распознаванию речи на малоресурсных языках, в частности тайском и вьетнамском [51], представили интеграцию БЯМ Qwen2.5 и Gemma3 и аудиокодера Whisper.

Интеграция на основе аудиотокенов. При этом подходе речевой сигнал преобразуется в дискретные единицы, которые затем подаются на вход БЯМ аналогично текстовым токенам. В работе [52] предложена модель SpeechGPT, предназначенная для распознавания и генерации речи. Она состоит из речевого кодера в дискретные токены (использующего HuBERT), расширенной БЯМ LLaMA, в словарь которой добавлены речевые токены, и речевого декодера, который преобразует речевые токены обратно в аудиосигнал. Дискретные токены также используются в мультимодальной модели AudioPaLM [53], которая обрабатывает и генерирует текст и речь, выполняя распознавание, перевод и голосовой перевод. Авторы исследовали модели на основе PaLM [17] и PaLM 2 [54] с 8 млрд параметров, предобученные только на текстовых данных. В работе [55] предлагаются так называемые дискретные речевые единицы, генерируемые из скрытого представления речевого кодера путем кластеризации, которые преобразуются речевым адаптером в векторные представления БЯМ, которые конкатенируются с текстовым запросом. Преимущество такого подхода — в применимости к задачам не только распознавания, но и синтеза речи. К недостаткам относятся высокие вычислительные требования и меньшая точность по сравнению с интеграцией на основе скрытых представлений [43].

Стоит отметить, что для распознавания речи предпочтительна интеграция на основе скрытых представлений, тогда как для синтеза — на основе токенов. Примером комбинированного подхода является LauraGPT [56], которая кодирует входное аудио в непрерывные представления, а выходные данные генерирует из дискретных кодеков. В современных исследованиях эти подходы получают дальнейшее развитие с фокусом на оптимизации для различных сценариев. Например, в работе [57] представлен подход к распознаванию речи с переключением кодов (китайский-английский) на основе БЯМ Qwen2 7B с MoE и токеном прерывания, что обеспечивает согласование между генерацией текста и CPP.

Контекстуализация и контекстное обучение

Одной из важных особенностей БЯМ является способность учитывать контекст, т. е. извлекать контекстную информацию из входной последовательности при генерации выходной последовательности. Достаточно большое число исследований посвящено контекстуализации для улучшения текста, генерируемого БЯМ. В качестве примера можно привести работу [58], авторы которой предложили модель, названную Speech LLaMA. Модель состоит из двух компонентов:

аудиокодера, преобразующего речь в скрытые представления, и БЯМ-декодера (LLaMA 7B), порождающего текст на основе аудио и контекста (например, заголовка и описания видео). Эксперименты показали снижение значения WER на 6 %.

Исследования в области контекстуализации привели к возникновению контекстного обучения (in-context learning), которое чаще всего применяется для переоценки или корректировки гипотез распознавания в задачах распознавания речи. При этом, помимо гипотезы распознавания, которую необходимо откорректировать, на вход БЯМ подаются примеры исправлений, а также соответствующие запросы, что можно формализовать следующим образом [59]:

$$y = f(I, (x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k), x),$$

где y — исправленный вывод; $f(\cdot)$ — функция контекстного исправления гипотез распознавания, реализуемая БЯМ; I — запрос для БЯМ; x — гипотеза распознавания, подлежащая корректировке; $(x_i, y_i)_{i=1}^{k}$ — примеры исправления ошибок, где k — число примеров.

Например, контекстное обучение для корректировки результатов распознавания использовано в работе [59], посвященной применению различных версий GPT-3.5 и GPT-4 для работы с корпусом LibriSpeech и корпусом китайской речи Aishell-1. На вход БЯМ подавался результат распознавания и соответствующие запросы для исправления потенциальных ошибок. В работе рассмотрено несколько стратегий: введение запросов с варьированием степени детализации, обучение на одном, двух и трех примерах, несколько попыток коррекции с выбором результата с наименьшим WER. Тем не менее авторам не удалось снизить WER за счет применения БЯМ. Более подробные запросы, а также предоставление большего числа примеров повышали точность, однако получаемый WER был все равно выше исходного. Даже при разрешении нескольких попыток исправления (до пяти) с выбором лучшего результата исправления БЯМ все равно вносили больше ошибок.

В работе [60] выполнено сравнение контекстного обучения, дообучения и низкоранговой адаптации для исправления ошибок распознавания. Были рассмотрены T5, LLaMA, GPT-3.5. Не имевшие предварительных примеров БЯМ с малым количеством параметров не дали заметных улучшений при распознавании с использованием Whisper, но использование БЯМ с большим количеством параметров и представлением контекста от WavLM позволило значительно улучшить точность, особенно в зашумленном и малоресурсных контекстах.

В работе [61] предложен метод контекстного обучения, названный авторами задачно-ориентированными запросами (task-activating prompting). Отличие от традиционного контекстного обучения состоит в том, что контекстное обучение происходит за один этап, состоящий из запроса, примеров и текста, подаваемого на вход. Задачно-ориентированные запросы — это многоэтапный процесс, состоящий из последовательности вопросов и ответов. Например, модель сначала спрашивают, знает ли она, что такое автоматическое распознавание речи, затем просят привести пример исправления ошибок и только после этого дают конкретные данные для обработки. В результате применения этого метода авторы смогли добиться уменьшения метрики WER на 31–38 %.

Обсуждение

Проведен сравнительный анализ методов применения БЯМ для распознавания речи по относительному сокращению WER (таблица). В части представленных работ оценка распознавания проводилась по показателю количества неправильно распознанных символов (character error rate, CER).

Из таблицы видно, что применение БЯМ для переранжирования гипотез демонстрирует стабильное улучшение результатов распознавания, но в то же время во многих случаях наблюдаемое улучшение лишь незначительно превосходит показатели эталонных систем, не использующих БЯМ. Применение БЯМ для коррекции ошибок часто демонстрирует улучшение результатов, при этом задача выполняется несколько быстрее, чем при переранжировании. Генеративный характер БЯМ позволяет им порождать исправленный текст напрямую, минуя фазу многоэтапного ранжирования, что обеспечивает существенное повышение скорости обработки. Однако, несмотря на очевидные преимущества в эффективности и скорости, серьезной проблемой в данном случае остается риск галлюцинаций, в принципе присущий генеративным моделям. В контексте коррекции ошибок это может означать не только неспособность исправить существующую ошибку, но и внесение новой, некорректной информации в текст. Работы с применением контекстуализации еще больше разнятся по результатам, демонстрируя как значительные улучшения по сравнению с базовыми моделями, так и ухудшения. Этот разброс может быть связан со сложностью применения БЯМ на основе декодировщиков и необходимостью тщательного подбора запросов, однако при условии использования неглубоких моделей и детальной контекстуализации дан-

■ Сравнительный анализ методов применения БЯМ для распознавания речи
■ Comparative analysis of LLM application methods for ASR tasks

Ссылка	Архитектура CPP	Архитектура БЯМ	Речевой корпус	Относительное сокращение WER, %
Переранжирование гипотез распознавания				
[30]	Listen, Attend and Spell (LAS)	BERT	LibriSpeech Clean	22,18
			LibriSpeech Other	14,73
[31]	TDNN/HMM	BERT	AMI	4,39
[32]	Whisper large v2	330M, аналогичная OPT	LibriSpeech Clean/Other	17,70/12,92
		7B, аналогичная Llama		18,14/14,79
Коррекция ошибок распознавания				
[33]	FSMN + 3-граммная языковая модель	BART	Собственный корпус путунхуа	CER: 21,85
[36]	Conformer-Transducer	T5	LibriSpeech Clean	12,15
			LibriSpeech Other	11,19
[34]	Conformer-Transducer	ChatGPT	LibriSpeech Other	10,14
	Whisper			5,41
[37]	Conformer-Transducer	GPT-4	LibriSpeech Other	31,59
	Whisper Small.en			–2,83
[38]	MMS	Qwen1.5	SPREDS-U1 (20 языков) (ast-astrec.nict.go.jp/en/release/SPREDS-U1)	CER: 70,16 (англ.)/ 31,70 (рус.)
	OWSM v3.1			CER: 34,65 (англ.)/ 46,45 (рус.)
	Whisper v3			CER: 44,19 (англ.) / 5,95 (рус.)
[39]	Комбинация моделей Whisper (различных версий), MMS, OWSM v3.1	ELYZA 7B, Qwen1.5 7B	SPREDS-U1-ja (японский)	39,81 (одной моделью)
				45,24
			CSJ (японский) [62]	6,67
Использование контекстного обучения				
[59]	Гибридная архитектура CTC/attention (предобученные веса от Wenet)	GPT-3.5 (разные версии), GPT-4	LibriSpeech Clean	–374,62
			LibriSpeech Other	–31,01
			Aishell-1 (китайский)	–21,99
[60]	WavLM, Whisper	T5	Различные корпуса, например WSJ	40
		LLaMA		51,11
Объединение БЯМ с аудиокодировщиком				
[51]	Whisper large-V3	Qwen2.5	Тайский	–22,14
			Вьетнамский	12,52
		Gemma3	Тайский	8,56
			Вьетнамский	11,63
[29]	Whisper	LLaDA 8B Instruct	LibriSpeech Other	12,3
[57]	Whisper-large-V3	Qwen2 7B с MoE и IDIT	Китайский с переключением на английский	19,83

ный подход может демонстрировать значительно лучшие результаты, чем прочие. Применение аудио-БЯМ к материалу малоресурсных языков показывает как улучшение, так и ухудшение результатов. Стоит отметить, однако, что использование многоязычной БЯМ, оснащенной механизмом «смеси экспертов», позволило существенно снизить значение показателя WER при распознавании китайской речи с переключением на английский. Этот факт указывает на перспективность дальнейших исследований по применению БЯМ для распознавания речи с переключением кодов в малоресурсных языках.

Проведенный анализ работ показывает, что некоторые архитектуры БЯМ, в частности рекуррентные и диффузионные, пока что не получили широкого распространения в контексте задач по распознаванию речи (несмотря на устойчивость последних к шумам). В целом наблюдается тенденция к использованию моделей, основанных на архитектуре декодера. Эта тенденция в свою очередь связана с рядом проблем, включая склонность этих моделей к галлюцинациям и их направленность на форматирование текста в соответствии с письменными нормами. Последнее обстоятельство может существенно затруднять точную передачу сказанного «слово в слово».

Кроме того, БЯМ могут использоваться совместно с СРР не только для повышения точности распознавания, но также для предобработки данных и постобработки результатов распознавания. В частности, способность БЯМ выполнять генерацию текстов может использоваться для аугментации текстовых данных для последующего обучения языковых моделей, что может быть особенно полезным при создании СРР для малоресурсных языков и речи с переключением кодов [63]. Постобработка результатов распознавания с помощью БЯМ может заключаться в резюмировании распознанного текста [64], переводе на другой язык [65], генерации ответов на речевые запросы пользователя [66], диаризации речи дикторов [67].

Также не трудно заметить, что различные методы применения БЯМ для распознавания речи в российских исследованиях представлены мало, хотя имеются подобные работы по применению акустических моделей на основе трансформера для распознавания русской речи [68] и моделей с интеграцией на основе латентных представлений аудиокодера и языковой модели BERT для распознавания карельской речи [50]. В контексте постобработки результатов распознавания с помощью БЯМ российскими учеными также рассматривалась задача генерации ответов на речевые запросы пользователя, например, в работе [69] описывается применение BERT и GPT-2 для этих задач.

Заключение

В настоящей статье систематизированы и обобщены существующие способы применения больших языковых моделей в СРР. Особое внимание уделено использованию БЯМ для переоценки списка гипотез, коррекции ошибок, а также различным способам интеграции аудиоданных в БЯМ.

Анализ показал, что БЯМ действительно обладают потенциалом к значительному улучшению результатов распознавания — за счет эффективных механизмов переранжирования гипотез или прямой коррекции ошибок. Однако наблюдаемое улучшение не всегда является кардинальным по сравнению с эталонными системами, не использующими БЯМ. Кроме того, применение генеративных БЯМ сопряжено с такими проблемами, как галлюцинации и избыточные коррекции при порождении письменного представления текста, что безусловно влияет на точность распознавания устной речи. Тем не менее разнообразие методов интеграции аудиоданных и БЯМ — от каскадных до основанных на скрытых представлениях и аудиотокенах, при том, что каждый из них имеет свои преимущества и недостатки — является самим по себе мощным инструментом для дальнейшего повышения эффективности БЯМ в задачах распознавания речи.

В целом, несмотря на уже достигнутые успехи, использование БЯМ в распознавании речи все еще находится на стадии активного развития. Применение новейших архитектур БЯМ, в частности рекуррентных и диффузионных моделей, которые обладают повышенной устойчивостью к шуму, представляется весьма перспективным направлением. Дальнейшие исследования могут быть связаны с улучшением методов, позволяющих эффективно контролировать генерацию БЯМ для минимизации галлюцинаций и обеспечения точности передачи устной речи, а также с исследованием потенциала БЯМ в контексте малоресурсного распознавания и переключения кодов.

Финансовая поддержка

Данное исследование выполнено в рамках бюджетной темы СПб ФИЦ РАН (№ FFZF-2025-0003).

Литература

1. **Zhao W. X., Zhou K., Li J., Tang T., Wang X., Hou Y., Min Y., Zhang B., Zhang J., Dong Z., Du Y., Yang C., Chen Y., Chen Z., Jiang J., Ren R., Li Y., Tang X., Liu Z., Liu P., Nie J. Y., Wen J. R.** A Survey of large language models. *arXiv preprint*, 2023. arXiv:2303.18223. doi:10.48550/arXiv.2303.18223
2. **Minaee Sh., Mikolov T., Nikzad N., Chenaghlu M. A., Socher R., Amatriain X., Gao J.** Large language models: A survey. *arXiv preprint*, 2024. arXiv:2402.06196. doi:10.48550/arXiv.2402.06196
3. **Wang F., Zhang Z., Zhang X., Wu Z., Mo T., Lu Q., Wang W., Li R., Xu J., Tang X., He Q., Ma Y., Huang M., Wang S.** A comprehensive survey of small language models in the era of large language models: Techniques, enhancements, applications, collaboration with LLMs, and trustworthiness. *arXiv preprint*, 2024. arXiv:2411.03350. doi:10.48550/arXiv.2411.03350
4. **Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser Ł., Polosukhin I.** Attention is all you need. *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS-2017)*, 2017, pp. 6000–6010. doi:10.48550/arXiv.1706.03762
5. **Капуста К. Л., Кипяткова И. С., Кагиров И. А.** Аналитический обзор интегральных моделей и стратегий распознавания речи на основе архитектуры трансформер. *Информационно-управляющие системы*, 2024, № 5, с. 2–15. doi:10.31799/1684-8853-2024-5-2-15, EDN: MWTGXE
6. **Hwang S., Lahoti A., Puduppully R., Dao T., Gu A.** Hydra: Bidirectional state space models through generalized matrix mixers. *Proceedings of the 38th Annual Conference on Neural Information Processing Systems (NIPS-2024)*, pp. 110876–110908. doi:10.48550/arXiv.2407.09941
7. **Xu W., Hu W., Wu F., Sengamedu S.** DeTiME: Diffusion-enhanced topic modeling using encoder-decoder based LLM. *Findings of the Association for Computational Linguistics (EMNLP-2023)*, 2023, pp. 9040–9057. doi:10.18653/v1/2023.findings-emnlp.606
8. **Lewis M., Liu Y., Goyal N., Ghazvininejad M., Mohamed A., Levy O., Stoyanov V., Zettlemoyer L.** BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 7871–7880. doi:10.18653/v1/2020.acl-main.703
9. **Liu Y., Gu J., Goyal N., Li X., Edunov S., Ghazvininejad M., Lewis M., Zettlemoyer L.** Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 2020, vol. 8, pp. 726–742. doi:10.1162/tacl_a_00343
10. **Raffel C., Shazeer N., Roberts A., Lee K., Narang Sh., Matena M., Zhou Y., Li W., Liu P. J.** Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 2020, vol. 21, iss. 1, pp. 5485–5551. doi:10.48550/arXiv.1910.10683
11. **Xue L., Constant N., Roberts A., Kale M., Al-Rfou R., Siddhant A., Barua A., Raffel C.** mT5: A massively multilingual pre-trained text-to-text transformer. *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-2021): Human Language Technologies*, 2021, pp. 483–498. doi:10.18653/v1/2021.naacl-main.41
12. **Devlin J., Chang M. W., Lee K., Toutanova K.** BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-2021): Human Language Technologies*, 2021, pp. 4171–4186. doi:10.18653/v1/N19-1423
13. **Kuratov Yu., Arkhipov M.** Adaptation of deep bidirectional multilingual transformers for Russian language. *arXiv preprint*, 2019. arXiv:1905.07213. doi:10.48550/arXiv.1905.07213
14. **Liu Y., Ott M., Goyal N., Du J., Joshi M., Chen D., Levy O., Lewis M., Zettlemoyer L., Stoyanov V.** RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint*, 2019. arXiv:1907.11692. doi:10.48550/arXiv.1907.11692
15. **Conneau A., Khandelwal K., Goyal N., Chaudhary V., Wenzek G., Guzmán F., Grave E., Ott M., Zettlemoyer L., Stoyanov V.** Unsupervised cross-lingual representation learning at scale. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL-2020)*, 2020, pp. 8440–8451. doi:10.18653/v1/2020.acl-main.747
16. **Radford A., Narasimhan K., Salimans T., Sutskever I.** Improving language understanding by generative pre-training. *OpenAI Blog*, 2018. https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf (дата обращения: 14.05.2025).
17. **Chowdhery A., Narang S., Devlin J., Bosma M., et al.** PaLM: Scaling language modeling with pathways. *The Journal of Machine Learning Research*, 2023, vol. 24, iss. 1, pp. 11324–11436. doi:10.48550/arXiv.2204.02311
18. **Touvron H., Lavril T., Izacard G., Martinet X., Lachaux M. A., Lacroix T., Rozière B., Goyal N., Hambro E., Azhar F., Rodriguez A., Joulin A., Grave E., Lample G.** Llama: Open and efficient foundation language models. *arXiv preprint*, 2023. arXiv:2302.13971. doi:10.48550/arXiv.2302.13971
19. **Almazrouei E., Alobeidli H., Alshamsi A., Cappelli A., Cojocaru R., Debbah M., Goffinet E., Heslow D., Launay J., Malartic Q., Mazotta D., Noune B., Pannier B., Penedo G.** The Falcon series of open language models. *arXiv preprint*, 2023. arXiv:2311.16867. doi:10.48550/arXiv.2311.16867

20. **Jiang D., Wu B., Chen C., Li R., Chen G., Sun Y., Kong X., Li L.** From clip to dino: Visual encoders shout in multi-modal large language models. *arXiv preprint*, 2023. arXiv:2310.08825. doi:10.48550/arXiv.2310.08825
21. **Jiang A. Q., Sablayrolles A., Roux A., Mensch A., Savary B., Bamford C., Chaplot D. S., de las Casas D., Hanna E. B., Bressand F., Lengyel G., Bour G., Lample G., Lavaud L. R., Saulnier L., Lachaux M.-A., Stock P., Subramanian S., Yang S., Antoniak S., Le Scao T., Gervet T., Lavril T., Wang T., Lacroix T., El Sayed W.** Mixtral of experts. *arXiv preprint*, 2024. arXiv:2401.04088. doi:10.48550/arXiv.2401.04088
22. **Fedus W., Zoph B., Shazeer N.** Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *The Journal of Machine Learning Research*, 2022, vol. 23, iss. 1, pp. 1–39. doi:10.48550/arXiv.2101.03961
23. **Gu A., Dao T.** Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint*, 2023. arXiv:2312.00752. doi:10.48550/arXiv.2312.00752
24. **Peng B., Alcaide E., Anthony Q., Albalak A., Arcadinho S., Biderman S., Cao H., Cheng X., Chung M., Grella M., Kiran G. K., He X., Hou H., Lin J., Kazienko P., Kocon J., Kong J., Koptyra B., Lau H., Mantri K. S. I., Mom F., Saito A., Song G., Tang X., Wang B., Wind J. S., Wozniak S., Zhang R., Zhang Z., Zhao Q., Zhou P., Zhou Q., Zhu J., Zhu R.-J.** RWKV: Reinventing RNNs for the transformer era. *Findings of the Association for Computational Linguistics (EMNLP-2023)*, 2023, pp. 14048–14077. doi:10.18653/v1/2023.findings-emnlp.936
25. **Beck M., Pöppel K., Spanring M., Auer A., Prudnikova O., Kopp M., Klambauer G., Brandstetter J., Hochreiter S.** xLSTM: Extended long short-term memory. *Advances in Neural Information Processing Systems*, 2025, vol. 37, pp. 107547–107603. doi:10.48550/arXiv.2405.04517
26. **De S., McLeish T., Botev A., Gu A., Dao T., Goyal N.** Griffin: Mixing gated linear recurrences with local attention for efficient language models. *arXiv preprint*, 2024. arXiv:2402.19427. doi:10.48550/arXiv.2402.19427
27. **Li X., Thickstun J., Gulrajani I., Liang P. S., Hashimoto T. B.** Diffusion-LM improves controllable text generation. *Advances in Neural Information Processing Systems*, 2022, vol. 35, pp. 4328–4343. doi:10.48550/arXiv.2205.14217
28. **Nie S., Zhu F., You Z., Zhang X., Ou J., Hu J., Li C.** Large language diffusion models. *Proceedings of Workshop on Deep Generative Model in Machine Learning: Theory, Principle and Efficacy (ICLR-2025)*, 2025. doi:10.48550/arXiv.2501.11720
29. **Wang M., Liu Zh., Jin Z., Sun G., Zhang Ch., Woodland P. C.** Audio-conditioned diffusion LLMs for ASR and deliberation processing. *arXiv preprint*, 2025. arXiv:2509.16622. doi:10.48550/arXiv.2509.16622
30. **Shin J., Lee Y., Jung K.** Effective sentence scoring method using BERT for speech recognition. *Proceedings of Machine Learning Research*, 2019, pp. 1081–1093. doi:10.48550/arXiv.1910.09932
31. **Chiu S. H., Chen B.** Innovative BERT-based reranking language models for speech recognition. *IEEE Spoken Language Technology Workshop (SLT-2021)*, 2021, pp. 266–271. doi:10.1109/SLT48900.2021.9383578
32. **Shivakumar P. G., Kolehmainen J., Gourav A., Gu Y., Gandhe A., Rastrow A., Bulyko I.** Speech recognition rescoring with large speech-text foundation models. *Proceedings of 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2025)*, 2025, pp. 1–5. doi:10.1109/ICASSP48485.2025.10494321
33. **Li S., Ko Y., Ito A.** LLM as decoder: Investigating lattice-based speech recognition hypotheses rescoring using LLM. *Proceedings of 2024 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC-2024)*, 2024, pp. 1–5. doi:10.1109/APSIPAASC58517.2024.10373582
34. **Ma R., Qian M., Manakul P., Gales M., Knill K.** Can generative large language models perform ASR error correction? *arXiv preprint*, 2023. arXiv:2307.04172. doi:10.48550/arXiv.2307.04172
35. **Zhao Y., Yang X., Wang J., Gao Y., Yan C., Zhou Y.** BART based semantic correction for Mandarin automatic speech recognition system. *Proceedings of the 22nd Annual Conference of the International Speech Communication Association, Interspeech 2021*, 2021, pp. 2017–2021. doi:10.21437/Interspeech.2021-1023
36. **Ma R., Gales M. J., Knill K. M., Qian M.** N-best T5: Robust ASR error correction using multiple input hypotheses and constrained decoding space. *Proceedings of the 24th Annual Conference of the International Speech Communication Association, Interspeech 2023*, 2023, pp. 3267–3271. doi:10.21437/Interspeech.2023-2189
37. **Ma R., Qian M., Gales M., Knill K.** ASR error correction using large language models. *IEEE Transactions on Audio, Speech and Language Processing*, 2025, vol. 33, pp. 1389–1401. doi:10.1109/TASLPRO.2025.3551083
38. **Li S., Chen C., Kwok C. Y., Chu C., Cheng E. S., Kawai H.** Investigating ASR error correction with large language model and multilingual 1-best hypotheses. *Proceedings of the 25th Annual Conference of the International Speech Communication Association, Interspeech 2024*, 2024, pp. 1315–1319. doi:10.21437/Interspeech.2024-368
39. **Ko Y., Li S., Yang C. H. H., Kawahara T.** Benchmarking Japanese speech recognition on ASR-LLM setups with multi-pass augmented generative error correction. *arXiv preprint*, 2024. arXiv:2408.16180. doi:10.48550/arXiv.2408.16180
40. **Wu H., Wang W., Wan Y., Jiao W., Lyu M.** ChatGPT or Grammarly? Evaluating ChatGPT on grammatical

error correction benchmark. *arXiv preprint*, 2023. arXiv:2303.13648. doi:10.48550/arXiv.2303.13648

41. **Mirbeygi M., Beigy H.** Prompt guided diffusion for controllable text generation. *Proceedings of the Tenth Workshop on Noisy and User-generated Text*, 2025, pp. 78–84. doi:10.18653/v1/2025.wnut-1.9
42. **Li Y., Wang X., Cao S., Zhang Y., Ma L., Xie L.** A transcription prompt-based efficient audio large language model for robust speech recognition. *Proceedings of the 25th Annual Conference of the International Speech Communication Association, Interspeech 2024*, 2024, pp. 1905–1909. doi:10.21437/Interspeech.2024-968
43. **Yang Z., Chen X., Zhang H., Li Y., Wang Y., Yu D.** When large language models meet speech: A survey on integration approaches. *arXiv preprint*, 2025. arXiv:2502.19548. doi:10.48550/arXiv.2502.19548
44. **Huang R., Li M., Yang D., Shi J., Chang X., Ye Z., Watanabe S.** AudioGPT: Understanding and generating speech, music, sound, and talking head. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024, vol. 38, no. 21, pp. 23802–23804. doi:10.1609/aaai.v38i21.30570
45. **Hsu W. N., Bolte B., Tsai Y. H. H., Lakhotia K., Salakhutdinov R., Mohamed A.** Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2021, vol. 29, pp. 3451–3460. doi:10.1109/TASLP.2021.31222
46. **Huang W. C., Chen Z., Chuang P. Y., Harwath D., Glass J.** Speech recognition by simply fine-tuning BERT. *arXiv preprint*, 2021. arXiv:2102.00291. doi:10.48550/arXiv.2102.00291
47. **Wang M., Han W., Shafran I., Wu Z., Chiu C.-C., Cao Y., Wang Y., Chen N., Zhang Y., Soltau H., Rubenstein P., Zilka L., Yu D., Meng Z., Pundak G., Siddhartha N., Schalkwyk J., Wu Y.** SLM: Bridge the thin gap between speech and text foundation models. *Proceedings of 2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU-2023)*, 2023, pp. 1–8. doi:10.1109/ASRU57964.2023.10389703
48. **Zhang Y., Qin J., Park D. S., Han W., Chiu C. C., Pang R., Le Q. V., Wu Y.** Google USM: Scaling automatic speech recognition beyond 100 languages. *arXiv preprint*, 2023. arXiv:2303.01037. doi:10.48550/arXiv.2303.01037
49. **Bai Y., Chen J., Chen J., Chen W., Chen Z., Ding C., Dong L., Dong Q., Du Y., Gao K., Gao L., Guo Y., Han M., Han T., Hu W., Hu X., Hu Y., Hua D., Huang L., Huang M., Huang Y., Jin J., Kong F., Lan Z., Li T., Li X., Li Z., Lin Z., Liu R., Liu S., Lu L., Lu Y., Ma J., Ma S., Pei Y., Shen C., Tan T., Tian X., Tu M., Wang B., Wang H., Wang Y., Wang Y., Xia H., Xia R., Xie S., Xu H., Yang M., Zhang B., Zhang J., Zhang W., Zhang Y., Zhang Y., Zheng Y., Zou M.** SEED-ASR: Understanding diverse speech and contexts with LLM-based speech recognition. *arXiv preprint*, 2024. arXiv:2407.04675. doi:10.48550/arXiv.2407.04675
50. **Кипяткова И. С., Кагиров И. А., Долгушин М. Д.** Применение предварительно обученных многоязычных моделей для распознавания карельской речи. *Информатика и автоматизация*, 2025, № 24(2), с. 604–630. doi:10.15622/ia.24.2.9
51. **Nguyen T., Hoang L. V., Tran H. D.** Qwen vs. Gemma integration with Whisper: A comparative study in multilingual SpeechLLM systems. *Proceedings of Workshop on Multilingual Conversational Speech Language Model (MLC-SLM)*, 2025. doi:10.21437/MLCSLM.2025-10
52. **Zhang D., Li S., Zhang X., Zhan J., Wang P., Zhou Y., Qiu X.** SpeechGPT: Empowering large language models with intrinsic cross-modal conversational abilities. *Findings of the Association for Computational Linguistics (EMNLP-2023)*, 2023, pp. 15757–15773. doi:10.18653/v1/2023.findings-emnlp.1055
53. **Rubenstein P. K., Asawaroengchai C., Nguyen D. D., Bapna A., Borsos Z., Chaumont Quitry de F., Chen P., El Badawy D., Han W., Kharitonov E., Muckenhirn H., Padfield D., Qin J., Rozenberg D., Sainath T., Schalkwyk J., Sharifi M., Ramanovich T. M., Tagliasacchi M., Tudor A., Velimirović M., Vincent D., Yu J., Wang Y., Zayats V., Zeghidour N., Zhang Y., Zhang Zh., Zilka L., Frank Ch.** AudioPaLM: A large language model that can speak and listen. *arXiv preprint*, 2023. arXiv:2306.12925. doi:10.48550/arXiv.2306.12925
54. **Anil R., Dai A. M., Firat O., Johnson M., Lepikhin D., et al.** PaLM 2 technical report. *arXiv preprint*, 2023. arXiv:2305.10403. doi:10.48550/arXiv.2305.10403
55. **Shon S., Yang C. H. H., Lee H., Kim J., Kim S., Kim T., Lee H. Y.** DiscreteSLU: A large language model with self-supervised discrete speech units for spoken language understanding. *Proceedings of the 25th Annual Conference of the International Speech Communication Association, Interspeech 2024*, 2024, pp. 4154–4158. doi:10.21437/Interspeech.2024-1306
56. **Du Z., Wang J., Chen Q., Chu Y., Gao Z., Li Z., Zhang S.** LauraGPT: Listen, attend, understand, and regenerate audio with GPT. *arXiv preprint*, 2023. arXiv:2310.04673. doi:10.48550/arXiv.2310.04673
57. **Zhang F., Geng W., Huang H., Shan Y., Yi C., Qu H.** Boosting code-switching ASR with mixture of experts enhanced speech-conditioned LLM. *Proceedings of 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2025)*, 2025, pp. 1–5. doi:10.1109/ICASSP49660.2025.10890030
58. **Lakomkin E., Wu C., Fathullah Y., Kalinli O., Seltzer M. L., Fuegen C.** End-to-end speech recognition contextualization with large language models. *Proceedings of 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2024)*, 2024, pp. 12406–12410. doi:10.1109/ICASSP48485.2024.10446898
59. **Min Z., Wang J.** Exploring the integration of large language models into automatic speech recognition

systems: An empirical study. *International Conference on Neural Information Processing (ICONIP-2023)*, 2023, pp. 69–84. doi:10.1007/978-981-99-8181-6_6

60. **Chen C., Hu Y., Yang C. H. H., Siniscalchi S. M., Chen P. Y., Chng E. S.** HyPoradise: An open baseline for generative speech recognition with large language models. *Proceedings of the 37th International Conference on Neural Information Processing Systems*, 2023, pp. 31665–31688. doi:10.48550/arXiv.2309.15701
61. **Yang C. H. H., Gu Y., Liu Y. C., Ghosh S., Bulyko I., Stolcke A.** Generative speech recognition error correction with large language models and task-activating prompting. *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU-2023)*, 2023, pp. 1–8. doi:10.1109/ASRU57964.2023.10389673
62. **Maekawa K.** Corpus of spontaneous Japanese: Its design and evaluation. *Proceedings of ISCA/IEEE Workshop on Spontaneous Speech Processing and Recognition*, 2003, paper MMO2.
63. **Nagano T., Kurata G., Thomas S., Kuo H. K. J., Bolanos D., Jung H., Saon G.** LLM based text generation for improved low-resource speech recognition models. *Proceedings of 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2025)*, 2025, pp. 1–5. doi:10.1109/ICASSP49660.2025.10888566
64. **Shang H., Wang Z., Li J., Liu Y., Zhang Y., Li X.** An end-to-end speech summarization using large language model. *Proceedings of the 25th Annual Conference of the International Speech Communication Association, Interspeech 2024*, 2024, pp. 1950–1954. doi:10.21437/Interspeech.2024-1428
65. **Xu J., Li Y., Wang Z., Zhang Y., Li X., Chen X.** MooER: LLM-based speech recognition and translation models from Moore Threads. *arXiv preprint*, 2024. arXiv:2408.05101. doi:10.48550/arXiv.2408.05101
66. **Nachmani E., Levkovitch A., Hirsch R., Salazar J., Asawaroengchai C., Mariooryad S., Ramanovich M. T.** Spoken question answering and speech continuation using spectrogram-powered LLM. *12th International Conference on Learning Representations (ICLR-2024)*, 2024. doi:10.48550/arXiv.2305.15255
67. **Wang Q., Huang Y., Zhao G., Clark E., Xia W., Liao H.** DiarizationLM: Speaker diarization post-processing with large language models. *Proceedings of the 25th Annual Conference of the International Speech Communication Association, Interspeech 2024*, 2024, pp. 3754–3758. doi:10.21437/Interspeech.2024-2214
68. **Kutsakov A., Maximenko A., Gospodinov G., Bogomolov P., Minkin F.** GigaAM: Efficient self-supervised learner for speech recognition. *arXiv preprint*, 2025. arXiv:2506.01192.
69. **Маслюхин С. М.** Диалоговая система на основе устных разговоров с доступом к неструктурированной базе знаний. *Научно-технический вестник информационных технологий, механики и оптики*, 2023, т. 23, № 1, с. 88–95. doi:10.17586/2226-1494-2023-23-1-88-95

UDC 004.934.2
doi:10.31799/1684-8853-2026-1-19-35
EDN: DSRKFE

Analytical review of the application of large language models for automatic speech recognition

I. S. Kipyatkova[a], PhD, Associate Professor, Senior Researcher, orcid.org/0000-0002-1264-4458, kipyatkova@iias.spb.su
M. D. Dolgushin[a], Junior Researcher, orcid.org/0000-0002-4344-2330
I. A. Kagirov[a], Research Fellow, orcid.org/0000-0003-1196-1117
[a]St. Petersburg Federal Research Center of the Russian Academy of Science, 39, 14th Line, 199178, Saint-Petersburg, Russian Federation

Introduction: One of the trends in natural language processing is the increasing use of large language models. In speech recognition systems, large language models are replacing traditional language models due to their ability to account for broader context. **Purpose:** To systematize and generalize current methods of joint use of automatic speech recognition systems and large language models. **Results:** We identify the main trends in the implementation of large language models to speech recognition. The analysis demonstrates that the application of large language models for hypothesis reranking and error correction consistently improves recognition results, although this improvement is not always fundamental and carries the risk of generating unreliable information due to possible model hallucinations. We conclude that contextualization and in-context learning of large language models can both improve, and degrade recognition results. **Practical relevance:** The generalizations proposed can find practical application in the development of automatic speech recognition systems for various natural and low-resource languages, as well as for code-switched speech. **Discussion:** Recurrent and diffusion large language model architectures have not yet gained widespread use in speech recognition tasks but hold significant potential. A trend towards using decoder-only architectures has been noted, which, in turn, gives rise to the problems of hallucinations and of an orientation towards written norms in text generation.

Keywords — large language models, hypothesis reranking, error correction, in-context learning, automatic speech recognition.

For citation: Kipyatkova I. S., Dolgushin M. D., Kagirov I. A. Analytical review of the application of large language models for automatic speech recognition. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2026, no. 1, pp. 19–35 (In Russian). doi:10.31799/1684-8853-2026-1-19-35, EDN: DSRKFE

Financial support
This survey was financially supported by budgetary theme No. FFZF-2025-0003.

References

1. Zhao W. X., Zhou K., Li J., Tang T., Wang X., Hou Y., Min Y., Zhang B., Zhang J., Dong Z., Du Y., Yang C., Chen Y., Chen Z., Jiang J., Ren R., Li Y., Tang X., Liu Z., Liu P., Nie J. Y., Wen J. R. A survey of Large Language Models. *arXiv preprint*, 2023. arXiv:2303.18223. doi:10.48550/arXiv.2303.18223
2. Minaee Sh., Mikolov T., Nikzad N., Chenaghlu M. A., Socher R., Amatriain X., Gao J. Large language models: A survey. *arXiv preprint*, 2024. arXiv:2402.06196. doi:10.48550/arXiv.2402.06196
3. Wang F., Zhang Z., Zhang X., Wu Z., Mo T., Lu Q., Wang W., Li R., Xu J., Tang X., He Q., Ma Y., Huang M., Wang S. A comprehensive survey of small language models in the era of large language models: Techniques, enhancements, applications, collaboration with LLMs, and trustworthiness. *arXiv preprint*, 2024. arXiv:2411.03350. doi:10.48550/arXiv.2411.03350
4. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A. N., Kaiser Ł., Polosukhin I. Attention is all you need. *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS-2017)*, 2017, pp. 6000–6010. doi:10.48550/arXiv.1706.03762
5. Kapusta K. L., Kipyatkova I. S., Kagirov I. A. Analytical survey of transformer-based end-to-end speech recognition models and strategies. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2024, no. 5, pp. 2–15 (In Russian). doi:10.31799/1684-8853-2024-5-2-15, EDN: MWTGXE
6. Hwang S., Lahoti A., Puduppully R., Dao T., Gu A. Hydra: Bidirectional state space models through generalized matrix mixers. *Proceedings of the 38th Annual Conference on Neural Information Processing Systems (NIPS-2024)*, pp. 110876–110908. doi:10.48550/arXiv.2407.09941
7. Xu W., Hu W., Wu F., Sengamedu S. DeTiME: Diffusion-enhanced topic modeling using encoder-decoder based LLM. *Findings of the Association for Computational Linguistics (EMNLP-2023)*, 2023, pp. 9040–9057. doi:10.18653/v1/2023.findings-emnlp.606
8. Lewis M., Liu Y., Goyal N., Ghazvininejad M., Mohamed A., Levy O., Stoyanov V., Zettlemoyer L. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 7871–7880. doi:10.18653/v1/2020.acl-main.703
9. Liu Y., Gu J., Goyal N., Li X., Edunov S., Ghazvininejad M., Lewis M., Zettlemoyer L. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 2020, vol. 8, pp. 726–742. doi:10.1162/tacl_a_00343
10. Raffel C., Shazeer N., Roberts A., Lee K., Narang Sh., Matena M., Zhou Y., Li W., Liu P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 2020, vol. 21, iss. 1, pp. 5485–5551. doi:10.48550/arXiv.1910.10683
11. Xue L., Constant N., Roberts A., Kale M., Al-Rfou R., Siddhant A., Barua A., Raffel C. mT5: A massively multilingual pre-trained text-to-text transformer. *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-2021): Human Language Technologies*, 2021, pp. 483–498. doi:10.18653/v1/2021.naacl-main.41
12. Devlin J., Chang M. W., Lee K., Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-2021*): Human Language Technologies, 2021, pp. 4171–4186. doi:10.18653/v1/N19-1423
13. Kuratov Yu., Arkhipov M. Adaptation of deep bidirectional multilingual transformers for Russian language. *arXiv preprint*, 2019. arXiv:1905.07213. doi:10.48550/arXiv.1905.07213
14. Liu Y., Ott M., Goyal N., Du J., Joshi M., Chen D., Levy O., Lewis M., Zettlemoyer L., Stoyanov V. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint*, 2019. arXiv:1907.11692. doi:10.48550/arXiv.1907.11692
15. Conneau A., Khandelwal K., Goyal N., Chaudhary V., Wenzek G., Guzmán F., Grave E., Ott M., Zettlemoyer L., Stoyanov V. Unsupervised cross-lingual representation learning at scale. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL-2020)*, 2020, pp. 8440–8451. doi:10.18653/v1/2020.acl-main.747
16. Radford A., Narasimhan K., Salimans T., Sutskever I. Improving language understanding by generative pre-training. *OpenAI Blog*, 2018. Available at: https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf (accessed 14 May 2025).
17. Chowdhery A., Narang S., Devlin J., Bosma M., et al. *The Journal of Machine Learning Research*, 2023, vol. 24, iss. 1, pp. 11324–11436. doi:10.48550/arXiv.2204.02311
18. Touvron H., Lavril T., Izacard G., Martinet X., Lachaux M. A., Lacroix T., Rozière B., Goyal N., Hambro E., Azhar F., Rodriguez A., Joulin A., Grave E., Lample G. Llama: Open and efficient foundation language models. *arXiv preprint*, 2023. arXiv:2302.13971. doi:10.48550/arXiv.2302.13971
19. Almazrouei E., Alobeidli H., Alshamsi A., Cappelli A., Cojocaru R., Debbah M., Goffinet E., Heslow D., Launay J., Malartic Q., Mazotta D., Noune B., Pannier B., Penedo G. The Falcon series of open language models. *arXiv preprint*, 2023. arXiv:2311.16867. doi:10.48550/arXiv.2311.16867
20. Jiang D., Wu B., Chen C., Li R., Chen G., Sun Y., Kong X., Li L. From clip to dino: Visual encoders shout in multi-modal large language models. *arXiv preprint*, 2023. arXiv:2310.08825. doi:10.48550/arXiv.2310.08825
21. Jiang A. Q., Sablayrolles A., Roux A., Mensch A., Savary B., Bamford C., Chaplot D. S., de las Casas D., Hanna E. B., Bressand F., Lengyel G., Bour G., Lample G., Lavaud L. R., Saulnier L., Lachaux M.-A., Stock P., Subramanian S., Yang S., Antoniak S., Le Scao T., Gervet T., Lavril T., Wang T., Lacroix T., El Sayed W. Mixtral of experts. *arXiv preprint*, 2024. arXiv:2401.04088. doi:10.48550/arXiv.2401.04088
22. Fedus W., Zoph B., Shazeer N. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *The Journal of Machine Learning Research*, 2022, vol. 23, iss. 1, pp. 1–39. doi:10.48550/arXiv.2101.03961
23. Gu A., Dao T. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint*, 2023. arXiv:2312.00752. doi:10.48550/arXiv.2312.00752
24. Peng B., Alcaide E., Anthony Q., Albalak A., Arcadinho S., Biderman S., Cao H., Cheng X., Chung M., Grella M., Kiran G. K., He X., Hou H., Lin J., Kazienko P., Kocon J., Kong J., Koptyra B., Lau H., Mantri K. S. I., Mom F., Saito A., Song G., Tang X., Wang B., Wind J. S., Wozniak S., Zhang R., Zhang Z., Zhao Q., Zhou P., Zhou Q., Zhu J., Zhu R.-J. RWKV: Reinventing RNNs for the transformer era. *Findings of the Association for Computational Linguistics (EMNLP-2023)*, 2023, pp. 14048–14077. doi:10.18653/v1/2023.findings-emnlp.936
25. Beck M., Pöppel K., Spanring M., Auer A., Prudnikova O., Kopp M., Klambauer G., Brandstetter J., Hochreiter S. xLSTM: Extended long short-term memory. *Advances in Neural Information Processing Systems*, 2025, vol. 37, pp. 107547–107603. doi:10.48550/arXiv.2405.04517
26. De S., McLeish T., Botev A., Gu A., Dao T., Goyal N. Griffin: Mixing gated linear recurrences with local attention for efficient language models. *arXiv preprint*, 2024. arXiv:2402.19427. doi:10.48550/arXiv.2402.19427
27. Li X., Thickstun J., Gulrajani I., Liang P. S., Hashimoto T. B. Diffusion-LM improves controllable text generation. *Advances in Neural Information Processing Systems*, 2022, vol. 35, pp. 4328–4343. doi:10.48550/arXiv.2205.14217
28. Nie S., Zhu F., You Z., Zhang X., Ou J., Hu J., Li C. Large language diffusion models. *Proceedings of Workshop on Deep Generative Model in Machine Learning: Theory, Principle and Efficacy (ICLR-2025)*, 2025. doi:10.48550/arXiv.2501.11720
29. Wang M., Liu Zh., Jin Z., Sun G., Zhang Ch., Woodland P. C. Audio-conditioned diffusion LLMs for ASR and deliberation processing. *arXiv preprint*, 2025. arXiv:2509.16622. doi:10.48550/arXiv.2509.16622
30. Shin J., Lee Y., Jung K. Effective sentence scoring method using BERT for speech recognition. *Proceedings of Machine Learning Research*, 2019, pp. 1081–1093. doi:10.48550/arXiv.1910.09932
31. Chiu S. H., Chen B. Innovative BERT-based reranking language models for speech recognition. *IEEE Spoken Language Technology Workshop (SLT-2021)*, 2021, pp. 266–271. doi:10.1109/SLT48900.2021.9383578
32. Shivakumar P. G., Kolehmainen J., Gourav A., Gu Y., Gandhe A., Rastrow A., Bulyko I. Speech recognition rescoring with large speech-text foundation models. *Proceedings of 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2025)*, 2025, pp. 1–5. doi:10.1109/ICASSP48485.2025.10494321

33. Li S., Ko Y., Ito A. LLM as decoder: Investigating lattice-based speech recognition hypotheses rescoring using LLM. *Proceedings of 2024 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC-2024)*, 2024, pp. 1–5. doi:10.1109/APSIPAASC58517.2024.10373582
34. Ma R., Qian M., Manakul P., Gales M., Knill K. Can generative large language models perform ASR error correction? *arXiv preprint*, 2023. arXiv:2307.04172. doi:10.48550/arXiv.2307.04172
35. Zhao Y., Yang X., Wang J., Gao Y., Yan C., Zhou Y. BART based semantic correction for Mandarin automatic speech recognition system. *Proceedings of the 22nd Annual Conference of the International Speech Communication Association, Interspeech 2021*, 2021, pp. 2017–2021. doi:10.21437/Interspeech.2021-1023
36. Ma R., Gales M. J., Knill K. M., Qian M. N-best T5: Robust ASR error correction using multiple input hypotheses and constrained decoding space. *Proceedings of the 24th Annual Conference of the International Speech Communication Association, Interspeech 2023*, 2023, pp. 3267–3271. doi:10.21437/Interspeech.2023-2189
37. Ma R., Qian M., Gales M., Knill K. ASR error correction using large language models. *IEEE Transactions on Audio, Speech and Language Processing*, 2025, vol. 33, pp. 1389–1401. doi:10.1109/TASLPRO.2025.3551083
38. Li S., Chen C., Kwok C. Y., Chu C., Cheng E. S., Kawai H. Investigating ASR error correction with large language model and multilingual 1-best hypotheses. *Proceedings of the 25th Annual Conference of the International Speech Communication Association, Interspeech 2024*, 2024, pp. 1315–1319. doi:10.21437/Interspeech.2024-368
39. Ko Y., Li S., Yang C. H. H., Kawahara T. Benchmarking Japanese speech recognition on ASR-LLM setups with multi-pass augmented generative error correction. *arXiv preprint*, 2024. arXiv:2408.16180. doi:10.48550/arXiv.2408.16180
40. Wu H., Wang W., Wan Y., Jiao W., Lyu M. ChatGPT or Grammarly? Evaluating ChatGPT on grammatical error correction benchmark. *arXiv preprint*, 2023. arXiv:2303.13648. doi:10.48550/arXiv.2303.13648
41. Mirbeygi M., Beigy H. Prompt guided diffusion for controllable text generation. *Proceedings of the Tenth Workshop on Noisy and User-generated Text*, 2025, pp. 78–84. doi:10.18653/v1/2025.wnut-1.9
42. Li Y., Wang X., Cao S., Zhang Y., Ma L., Xie L. A transcription prompt-based efficient audio large language model for robust speech recognition. *Proceedings of the 25th Annual Conference of the International Speech Communication Association, Interspeech 2024*, 2024, pp. 1905–1909. doi:10.21437/Interspeech.2024-968
43. Yang Z., Chen X., Zhang H., Li Y., Wang Y., Yu D. When large language models meet speech: A survey on integration approaches. *arXiv preprint*, 2025. arXiv:2502.19548. doi:10.48550/arXiv.2502.19548
44. Huang R., Li M., Yang D., Shi J., Chang X., Ye Z., Watanabe S. AudioGPT: Understanding and generating speech, music, sound, and talking head. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2024, vol. 38, no. 21, pp. 23802–23804. doi:10.1609/aaai.v38i21.30570
45. Hsu W. N., Bolte B., Tsai Y. H. H., Lakhotia K., Salakhutdinov R., Mohamed A. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2021, vol. 29, pp. 3451–3460. doi:10.1109/TASLP.2021.31222
46. Huang W. C., Chen Z., Chuang P. Y., Harwath D., Glass J. Speech recognition by simply fine-tuning BERT. *arXiv preprint*, 2021. arXiv:2102.00291. doi:10.48550/arXiv.2102.00291
47. Wang M., Han W., Shafran I., Wu Z., Chiu C.-C., Cao Y., Wang Y., Chen N., Zhang Y., Soltau H., Rubenstein P., Zilka L., Yu D., Meng Z., Pundak G., Siddhartha N., Schalkwyk J., Wu Y. SLM: Bridge the thin gap between speech and text foundation models. *Proceedings of 2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU-2023)*, 2023, pp. 1–8. doi:10.1109/ASRU57964.2023.10389703
48. Zhang Y., Qin J., Park D. S., Han W., Chiu C. C., Pang R., Le Q. V., Wu Y. Google USM: Scaling automatic speech recognition beyond 100 languages. *arXiv preprint*, 2023. arXiv:2303.01037. doi:10.48550/arXiv.2303.01037
49. Bai Y., Chen J., Chen J., Chen W., Chen Z., Ding C., Dong L., Dong Q., Du Y., Gao K., Gao L., Guo Y., Han M., Han T., Hu W., Hu X., Hu Y., Hua D., Huang L., Huang M., Huang Y., Jin J., Kong F., Lan Z., Li T., Li X., Li Z., Lin Z., Liu R., Liu S., Lu L., Lu Y., Ma J., Ma S., Pei Y., Shen C., Tan T., Tian X., Tu M., Wang B., Wang H., Wang Y., Wang Y., Xia H., Xia R., Xie S., Xu H., Yang M., Zhang B., Zhang J., Zhang W., Zhang Y., Zhang Y., Zheng Y., Zou M. SEED-ASR: Understanding diverse speech and contexts with LLM-based speech recognition. *arXiv preprint*, 2024. arXiv:2407.04675. doi:10.48550/arXiv.2407.04675
50. Kipyatkova I., Kagirov I., Dolgushin M. Use of pretrained multilingual models for Karelian speech recognition. *Informatics and Automation*, 2025, no. 24(2), pp. 604–630 (In Russian). doi:10.15622/ia.24.2.9
51. Nguyen T., Hoang L. V., Tran H. D. Qwen vs. Gemma integration with Whisper: A comparative study in multilingual SpeechLLM systems Qwen vs. Gemma integration with Whisper: A comparative study in multilingual SpeechLLM systems. *Proceedings of Workshop on Multilingual Conversational Speech Language Model (MLC-SLM)*, 2025. doi:10.21437/MLCSLM.2025-10
52. Zhang D., Li S., Zhang X., Zhan J., Wang P., Zhou Y., Qiu X. SpeechGPT: Empowering large language models with intrinsic cross-modal conversational abilities. *Findings of the Association for Computational Linguistics (EMNLP-2023)*, 2023, pp. 15757–15773. doi:10.18653/v1/2023.findings-emnlp.1055
53. Rubenstein P. K., Asawaroengchai C., Nguyen D. D., Bapna A., Borsos Z., Chaumont Quitry de F., Chen P., El Badawy D., Han W., Kharitonov E., Muckenhirn H., Padfield D., Qin J., Rozenberg D., Sainath T., Schalkwyk J., Sharifi M., Ramanovich T. M., Tagliasacchi M., Tudor A., Velimirović M., Vincent D., Yu J., Wang Y., Zayats V., Zeghidour N., Zhang Y., Zhang Zh., Zilka L., Frank Ch. AudioPaLM: A large language model that can speak and listen. *arXiv preprint*, 2023. arXiv:2306.12925. doi:10.48550/arXiv.2306.12925
54. Anil R., Dai A. M., Firat O., Johnson M., et al. PaLM 2 technical report. *arXiv preprint*, 2023. arXiv:2305.10403. doi:10.48550/arXiv.2305.10403
55. Shon S., Yang C. H. H., Lee H., Kim J., Kim S., Kim T., Lee H. Y. DiscreteSLU: A large language model with self-supervised discrete speech units for spoken language understanding. *Proceedings of the 25th Annual Conference of the International Speech Communication Association, Interspeech 2024*, 2024, pp. 4154–4158. doi:10.21437/Interspeech.2024-1306
56. Du Z., Wang J., Chen Q., Chu Y., Gao Z., Li Z., Zhang S. LauraGPT: Listen, attend, understand, and regenerate audio with GPT. *arXiv preprint*, 2023. arXiv:2310.04673. doi:10.48550/arXiv.2310.04673
57. Zhang F., Geng W., Huang H., Shan Y., Yi C., Qu H. Boosting code-switching ASR with mixture of experts enhanced speech-conditioned LLM. *Proceedings of 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2025)*, 2025, pp. 1–5. doi:10.1109/ICASSP49660.2025.10890030
58. Lakomkin E., Wu C., Fathullah Y., Kalinli O., Seltzer M. L., Fuegen C. End-to-end speech recognition contextualization with large language models. *Proceedings of 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2024)*, 2024, pp. 12406–12410. doi:10.1109/ICASSP48485.2024.10446898
59. Min Z., Wang J. Exploring the integration of large language models into automatic speech recognition systems: An empirical study. *International Conference on Neural Information Processing (ICONIP-2023)*, 2023, pp. 69–84. doi:10.1007/978-981-99-8181-6_6
60. Chen C., Hu Y., Yang C. H. H., Siniscalchi S. M., Chen P. Y., Chng E. S. HyPoradise: An open baseline for generative speech recognition with large language models. *Proceedings of the 37th International Conference on Neural Information Processing Systems*, 2023, pp. 31665–31688. doi:10.48550/arXiv.2309.15701
61. Yang C. H. H., Gu Y., Liu Y. C., Ghosh S., Bulyko I., Stolcke A. Generative speech recognition error correction with large language models and task-activating prompting. *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU-2023)*, 2023, pp. 1–8. doi:10.1109/ASRU57964.2023.10389673
62. Maekawa K. Corpus of spontaneous Japanese: Its design and evaluation. *Proceedings of ISCA/IEEE Workshop on Spontaneous Speech Processing and Recognition*, 2003, paper MMO2.

63. Nagano T., Kurata G., Thomas S., Kuo H. K. J., Bolanos D., Jung H., Saon G. LLM based text generation for improved low-resource speech recognition models. *Proceedings of 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2025)*, 2025, pp. 1–5. doi:10.1109/ICASSP49660.2025.10888566
64. Shang H., Wang Z., Li J., Liu Y., Zhang Y., Li X. An end-to-end speech summarization using large language model. *Proceedings of the 25th Annual Conference of the International Speech Communication Association, Interspeech 2024*, 2024, pp. 1950–1954. doi:10.21437/Interspeech.2024-1428
65. Xu J., Li Y., Wang Z., Zhang Y., Li X., Chen X. MooER: LLM-based speech recognition and translation models from Moore Threads. *arXiv preprint*, 2024. arXiv:2408.05101. doi:10.48550/arXiv.2408.05101
66. Nachmani E., Levkovitch A., Hirsch R., Salazar J., Asawaroengchai C., Mariooryad S., Ramanovich M. T. Spoken question answering and speech continuation using spectrogram-powered LLM. *12th International Conference on Learning Representations (ICLR-2024)*, 2024. doi:10.48550/arXiv.2305.15255
67. Wang Q., Huang Y., Zhao G., Clark E., Xia W., Liao H. DiarizationLM: Speaker diarization post-processing with large language models. *Proceedings of the 25th Annual Conference of the International Speech Communication Association, Interspeech 2024*, 2024, pp. 3754–3758. doi:10.21437/Interspeech.2024-2214
68. Kutsakov A., Maximenko A., Gospodinov G., Bogomolov P., Minkin F. GigaAM: Efficient self-supervised learner for speech recognition. *arXiv preprint*, 2025. arXiv:2506.01192.
69. Masliukhin S. M. Dialogue system based on spoken conversations with access to an unstructured knowledge base. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2023, vol. 23, no. 1, pp. 88–95 (In Russian). doi:10.17586/2226-1494-2023-23-1-88-95

УДК 004.492.3
doi:10.31799/1684-8853-2026-1-36-47
EDN: LSYLOP

Научные статьи
Articles

Алгоритм автоматического построения регулярных выражений для предобработки журнальных сообщений произвольного формата в вычислительных системах

Г. А. Драчев [a,б], *соискатель, приглашенный преподаватель, руководитель проектов, orcid.org/0000-0003-1851-5507, pendal2@gmail.com*
[a] *Национальный исследовательский университет «Высшая школа экономики», Таллинская ул., 34, Москва, 123458, РФ*
[б] *Центр специальной системотехники — сервис, Варшавское ш., 71, Москва, 117556, РФ*

Введение: *предобработка журнальных сообщений необходима для структуризации журнальных сообщений и выделения полей-характеристик для последующего анализа в целях обнаружения аномалий и компьютерных атак в вычислительной системе. Наиболее приемлемый подход, который доминирует в коммерческих решениях, — построение регулярных выражений, соответствующих журнальным сообщениям, что требует трудоемкой ручной обработки.* ***Цель:*** *разработать алгоритм автоматического построения регулярных выражений журнальных сообщений в реальном масштабе времени, который можно применить к любому журнальному сообщению независимо от его формата и источника продуцирования.* ***Результаты:*** *анализ источников журнальных сообщений, способов их доставки в системы хранения и обработки, а также существующих форматов журнальных сообщений показал, что журнальные сообщения, даже в рамках одного формата, часто не стандартизованы по набору полей. Разработан алгоритм, позволяющий по тексту журнального сообщения сформировать соответствующий ему шаблон и на основе процедуры обработки шаблонов построить регулярное выражение с выделенными полями, т. е. структурировать журнальное сообщение. Помимо этого, спроектирована система хранения накопленных шаблонов журнальных сообщений и соответствующих им регулярных выражений. Для проведения экспериментальных исследований разработан программный комплекс, обеспечивающий построение регулярных выражений по текстам журнальных сообщений в автоматическом режиме. Программный комплекс апробирован на реальных вычислительных системах различных конфигураций.* ***Практическая значимость:*** *предложенный алгоритм позволяет структурировать журнальные сообщения произвольных типов и форматов. Структурированные журнальные сообщения могут быть использованы для расследования инцидентов информационной безопасности, аудита информационных систем, в качестве входных данных для анализаторов аномалий и компьютерных атак.*

Ключевые слова *— информационная безопасность, журналы вычислительной системы, журнальные сообщения, предобработка данных, структуризация журнальных сообщений, регулярное выражение.*

Для цитирования: Драчев Г. А. Алгоритм автоматического построения регулярных выражений для предобработки журнальных сообщений произвольного формата в вычислительных системах. *Информационно-управляющие системы*, 2026, № 1, с. 36–47. doi:10.31799/1684-8853-2026-1-36-47, EDN: LSYLOP

For citation: Drachev G. A. Algorithm for automatic construction of regular expressions for preprocessing arbitrary-format log messages in computing systems. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2026, no. 1, pp. 36–47 (In Russian). doi:10.31799/1684-8853-2026-1-36-47, EDN: LSYLOP

Введение

Вычислительные системы в ходе своей жизнедеятельности порождают огромный массив журнальных сообщений (ЖС) [1]. Источником ЖС в контуре вычислительной системы (ВС) может являться любой ее программный или аппаратный компонент. Транспортировка и хранение ЖС осуществляются различными протоколами ВС (SYSLOG, https://www.rfc-editor.org/rfc/rfc5424; SNMP, https://www.rfc-editor.org/rfc/rfc1902; журналируемыми файловыми системами и др. [2]). В работе [2] представлены примеры источников журнальных сообщений ВС.

Журнальное сообщение — это запись произвольного формата, которая отражает текущее изменение состояния одной из частей ВС. Каждая запись ЖС состоит из полей, несущих основную информационную нагрузку (временную метку, имя/идентификатор процесса, имя/идентификатор пользователя и др.), а также сопроводительных текстовых данных. В процессе журналирования ВС накапливается большое количество сообщений, которые позволяют анализировать поведение ВС и диагностировать возникающие ошибки, помогают в сопровождении ВС, в расследовании инцидентов информационной безопасности и компьютерных атак [3].

Журнальные сообщения по мере накопления записываются в файлы для их хранения (журналы). Для анализа журналов используются различные методы, например, машинного обучения [3, 4], анализа больших данных [5] и др. [6–8]. Абсолютное большинство из них не применимо для обработки ЖС в режиме реального

времени и (или) имеет ограничения по обработке форматов ЖС.

Единого стандарта для формирования ЖС нет, — программные и аппаратные источники формируют ЖС в различных форматах [1]. ЖС могут отличаться по набору, составу, количеству и смысловой нагрузке полей. Эти факторы не позволяют напрямую использовать рассмотренные методы обработки ЖС. Поэтому информацию из журнальных сообщений перед этапом анализа необходимо предварительно обработать — выделить поля, характеризующие ЖС, т. е. структурировать.

В научных работах предлагается множество подходов к решению задачи структуризации ЖС. Например, ручное формирование регулярных выражений (РВ) [2, 6] или правил разбора ЖС [7–10], анализ дерева фиксированной глубины [11], алгоритм кластеризации для создания шаблонов ЖС [12], метод построения FP-деревьев [13], алгоритм предобработки, основанный на выделении и анализе только одной «ключевой» части (характеристики) ЖС [5], статистические методы [14–17], алгоритмы машинного обучения [18–20], в частности большие языковые модели [21–24].

Существующим подходам структуризации ЖС присущ ряд функциональных недостатков:

— математические [12, 13] и статистические алгоритмы [14–17] не позволяют осуществлять обработку в режиме реального времени;

— алгоритмы машинного обучения не позволяют осуществлять качественную обработку ЖС произвольных форматов [18–20] либо не предназначены для обработки в режиме реального времени [21–24];

— алгоритмы, ориентированные на реализацию в режиме реального времени, ограничены

■ ***Таблица 1.*** Методы структуризации журнальных сообщений, применяемые в актуальных программных решениях
■ ***Table 1.*** Log structuring methods in modern software

Продукт	Основной метод структуризации	Недостатки метода структуризации
Elastic Stack (https://www.elastic.co/elastic-stack)	Регулярные выражения (шаблоны для языковой модели Grok)	Высокая сложность написания и отладки сложных шаблонов Регулярные выражения требуют точного соответствия формату. Любое отклонение приводит к ошибке структуризации ЖС Высокая нагрузка на процессор
Splunk (https://www.splunk.com/)	Конфигурирование правил разбора ЖС (props.conf) и поисковый язык (SPL)	Автоматическое извлечение полей производится с большим количеством лишних данных Ограничения по количеству извлекаемых полей Сложный алгоритм структуризации замедляет обработку в режиме реального времени
Graylog (https://graylog.org/)	Конфигурирование правил разбора ЖС	Неэффективен для произвольных форматов ЖС Затруднение отладки сложных цепочек правил Производительность падает при большом количестве правил
Grafana Loki (https://grafana.com/oss/loki/)	Описание меток (для выделения полей) и отложенная структуризация	Эффективность системы падает при увеличении количества и сложности описания меток Одновременная обработка ЖС произвольных форматов очень медленная
Vector (https://vector.dev/)	Декларативный язык (VRL) с использованием встроенных функций	Требуется изучение синтаксиса VRL для сложных преобразований Сложная отладка конфигурации
Fluentd / Bit (https://fluentbit.io/)	Плагины (на основе Regex, JSON и др.)	Производительность и возможности сильно зависят от выбранного плагина Использование множества плагинов усложняет конфигурацию и может привести к ошибкам структуризации, вплоть до нештатного завершения программы
Datadog (https://www.datadoghq.com/)	Шаблоны для языковой модели Grok	Интерфейс программы ограничивает возможности задания шаблонов Сложные нестандартные форматы все равно требуют ручного написания шаблонов
DeepLog (https://github.com/Thijs-vanede/DeepLog)	Предустановленные обработчики под журналы программного и аппаратного обеспечения от конкретных производителей	Плохая приспособленность для обработки уникальных форматов данных Зависимость от производителей компонентов ВС в обновлении и поддержке правил структуризации для нового оборудования/программного обеспечения

либо форматом ЖС [2, 7, 8, 10], либо количеством выделяемых характеристик [4, 6, 9, 11] и не позволяют структурировать редко встречающиеся ЖС.

В современных коммерческих продуктах доминирует подход к структуризации ЖС с помощью РВ или правил структуризации, составленных «вручную» (табл. 1).

Такой подход, с одной стороны, не ограничен одной выделяемой характеристикой ЖС, а с другой стороны, РВ можно подобрать для любого отдельного формата ЖС. Однако ручное формирование РВ не позволяет обрабатывать произвольные ЖС из-за разнообразия их форматов и содержания (из-за отсутствия единой стандартизации невозможно учесть все возможные варианты).

Для замены «ручного» формирования регулярного выражения предлагается разработать алгоритм автоматического построения регулярного выражения для произвольного журнального сообщения. В качестве входных данных используется журнальное сообщение произвольного формата от произвольного источника, в качестве выхода — сформированное регулярное выражение, которое соответствует журнальному сообщению, поданному на вход. Регулярное выражение должно содержать группы для структуризации журнального сообщения.

Формирование шаблонов журнальных сообщений

Большинство программных или аппаратных компонентов ВС генерирует массив ЖС различных форматов. Некоторые ЖС, например по протоколу SYSLOG, стандартизируют доставляемые сообщения при помощи добавления заголовка в начало сообщения, но не вносят изменений в неструктурированное содержание самих сообщений. Другие ЖС, например по протоколу SNMP, накладывают ограничения на формат сообщений. Сообщения по протоколу SNMP представляют собой последовательность OID [3] ключей и их значений через разделитель «;». Однако деревья OID ключей не стандартизированы и могут отличаться даже в рамках аналогичных источников ЖС (https://www.cisco.com/c/en/us/support/docs/smb/switches/Cisco-Business-Switching/kmgmt3636-snmpv3-common-oids-cbs350.html и https://community.cisco.com/t5/networking-knowledge-base/oid-list/ta-p/3117547). Отдельные журналы, например, генерируемые аппаратно-программным модулем доверенной загрузки (АПМДЗ, https://www.udcs.ru/catalog/sredstva-zashchity-informatsii/zashchita-ot-nesanktsionirovannogo-dostupa/apparatno-programmnyy-modul-doverennoy-zagruzki-apmdz-sobol/) или системой PARSEC (https://wiki.astralinux.ru/pages/viewpage.action?pageId = 67112737), имеют свои уникальные форматы ЖС и не используют возможности операционной системы для ведения журналов.

Однако вне зависимости от формата каждое ЖС представляет собой последовательность символов, которая запрограммирована заранее, исключая параметры, которые передаются в строку при ее продуцировании [8]. Вместо последовательностей символов можно рассматривать каждое ЖС как последовательность слов. Слово — это цепочка символов, ограниченная символами-разделителями.

Эмпирическим путем были выделены множества символов-разделителей (https://www.pcre.org/pcre2.txt) между словами ЖС:

[:space:] — множество управляющих символов, обозначающих разрыв между словами;

[:blank:] — множество пробельных символов;

[:punct:] — множество знаков пунктуации.

Цепочка символов, заключенная в кавычки, одинарные или двойные, учитывается как одно слово целиком, вне зависимости от наличия символов-разделителей внутри цепочки.

Назовем совпадающую по смыслу последовательность слов и символов-разделителей шаблоном ЖС. Формирование шаблона ЖС — процесс разбиения записи ЖС на слова и символы-разделители. Шаблон уже можно использовать как РВ, однако поиск по шаблону не «среагирует» на следующее ЖС того же типа, если в ЖС поменяется хотя бы одно слово. Отдельные слова у разных экземпляров одного типа ЖС могут отличаться (например, “user1” и “user2” (рис. 1)). Эти слова могут содержать важную информацию, характеризующую событие, описанное в ЖС:

— временную метку, когда было зафиксировано событие;

— субъект, который совершил или спровоцировал журналируемое действие;

— объект, над которым было совершено действие или изменение его состояния;

— результат журналируемого воздействия и др.

На рис. 1 представлен пример двух шаблонов ЖС одного типа, доставленных по протоколу SYSLOG от PAM модуля (https://uchet-jkh.ru/i/nastroika-pam-v-astra-linux/) (подключаемого модуля аутентификации) Unix-подобных ОС. ЖС содержат информацию об успешном открытии сессии по протоколу SSH в разное время для пользователей user1 и user2. В шаблонах выделены слова (поля), характеризующие событие ЖС.

Для определения полей-характеристик ЖС необходимо преобразовать его шаблон к общему виду, который будет подходить для всех ЖС того же типа.

<86>	Aug 31 15:20:21	dr-1 sshd[26272]:pam_unix(sshd:session): session opened for	user 1	user by (uid= 0)
<86>	Aug 31 18:36:51	dr-1 sshd[26272]:pam_unix(sshd:session): session opened for	user 2	user by (uid= 1)

общая часть поля-характеристики

■ ***Рис. 1.*** Два шаблона журнальных сообщений
■ ***Fig. 1.*** Two log message templates

Анализ и преобразование шаблонов журнальных сообщений

Для определения принадлежности шаблонов ЖС к одному типу и их преобразования к шаблону общего вида предложен механизм (анализатор шаблонов), который сравнивает новый шаблон с теми, которые были получены ранее. Для операции сравнения выбираются шаблоны совпадающей длины (определяется в словах) и подпоследовательности символов-разделителей. Результат работы анализатора шаблонов — это решение о внесении в базу шаблонов нового шаблона или изменении (выделении полей), при необходимости, одного из старых (т. е. приведение к общему виду). Если для нового шаблона нет совпадения в базе старых шаблонов, то он вносится как новая запись. Если находится шаблон (1) (рис. 2), который отличается от нового (2) не больше чем на половину слов от длины шаблона, то оба шаблона описывают один и тот же тип ЖС и должны быть преобразованы в шаблон общего вида (3). Если можно выделить набор слов, не совпадающий в сравниваемых шаблонах одного типа ЖС, то в шаблоне общего вида (3) слова такого набора будем считать изменяющимися полями-характеристиками (обозначим каждый из них символом «*»). Преобразование шаблонов (1) и (2) в шаблон общего вида (3) с выделением слов-характеристик представлен на рис. 2.

Поле Wed Feb 1 06:53:59 2023, представляющее собой временную метку ЖС, заключено в одинарные кавычки и поэтому воспринимается одним словом (см. рис. 2). Однако чаще всего (например, в SYSLOG, АПМДЗ и др.) временная метка не закрывается кавычками, что приводит к разбиению временной метки на отдельные поля. Чтобы сохранить целостность временной метки, предлагается использовать готовое решение: определять временную метку в ЖС при помощи функции из библиотеки timeGrinder (https://pkg.go.dev/github.com/gravwell/gravwell/v3/timegrinder#Extract) для ее обработки как целостного поля-характеристики.

Шаблон общего вида, полученный в результате работы анализатора шаблонов, описывает ЖС одного типа и позволяет определить поля-характеристики этих ЖС. Шаблон общего вида можно преобразовать в РВ при помощи замены «*» на «(.*)» (в РВ — любая последовательность символов), т. е. трансформировать конструкцию в группу (https://www.pcre.org/pcre2.txt).

Алгоритм построения регулярных выражений

Сконструируем формальный алгоритм построения РВ на основе ЖС произвольных форматов.

Журнальное сообщение 1:													
[f]'Wed Feb 1 06:53:59 2023' '/bin/atsrv' <21588,12087,0,0,0> [s] open("/usr/argusids/lib64/libtcryptf.so",NO_PERMS | O_LARGEFILE) = 0													
Журнальное сообщение 2:													
[f]'Wed Feb 1 06:54:04 2023' '/bin/argsh0' <21599,12087,0,0,0> [s] open("/bin/atsrv",NO_PERMS | O_LARGEFILE) = 0													
Шаблон (1) (для сообщения 1):													
f	Wed Feb 1 06:53:59 2023	/bin/atsrv	21588	12087	0	0	0	s	open	/usr/argusids/lib64/libtcryptf.so	NO_PERMS	O_LARGEFILE	0
Шаблон (2) (для сообщения 2):													
f	Wed Feb 1 06:54:04 2023	/bin/argsh0	21599	12087	0	0	0	s	open	/bin/atsrv	NO_PERMS	O_LARGEFILE	0
Шаблон общего вида (3):													
f	*	*	*	12087	0	0	0	s	open	*	NO_PERMS	O_LARGEFILE	0

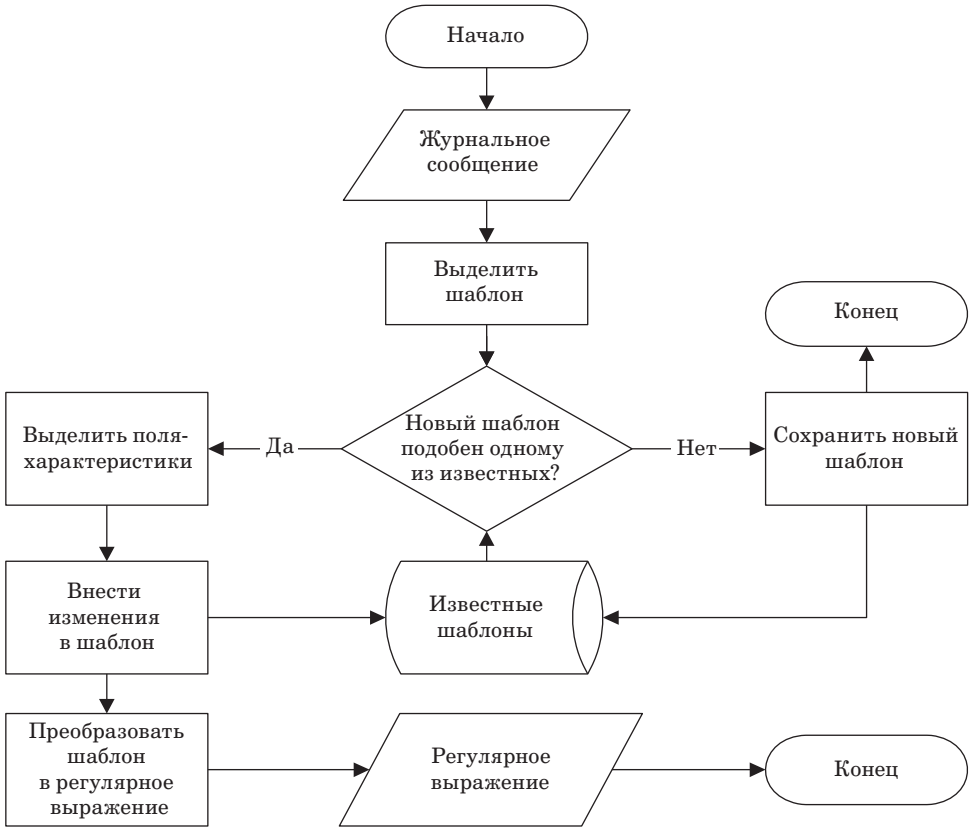
■ ***Рис. 2.*** Преобразование шаблона (1) в (3) по результатам сравнения с шаблоном (2)
■ ***Fig. 2.*** Conversion of template (1) to (3) after matching against template (2)

Шаг 1. По одному из протоколов $p \in P$, где P — протоколы транспортировки сообщений, на вход поступает информация $m_n^{s,p} \in M$, где M — информация в виде цепочки символов $\left\{m_1^{s,p}, ..., m_N^{s,p}\right\}$, N — максимальное возможное количество вариантов информации от известных в рамках конфигурации ВС источников $s \in S$, S — источники $\{s_1, ..., s_A\}$, где A — количество источников информации в ВС.

Шаг 2. Полученная цепочка $m_n^{s,p}$ разбивается на последовательность слов $\left(\left(w_q\right)_1^Q\right)^{m_n^{s,p}}$, т. е. формируется шаблон, где Q — количество слов в цепочке. Разбиение осуществляется путем поиска разделителей $t \in T$, $T = \{t_1, ..., t_Z\}$, $1 \le z \le Z$, где Z — количество вариантов символов-разделителей. Обозначим подпоследовательность символов-разделителей для шаблона $\left(\left(w_q\right)_1^Q\right)^{m_n^{s,p}}$ как $T_Q^{m_n^{s,p}}$.

Шаг 3. Новый шаблон $\left(\left(w_q\right)_1^Q\right)^{m_n^{s,p}}$ сравнивается с уже известными шаблонами из множества шаблонов общего вида $W = \left\{\left(\left(w_y\right)_1^Y\right)^{m_1^{s,p}}, ..., \left(\left(w_y\right)_1^Y\right)^{m_N^{s,p}}\right\}$, у которых такая же длина $Q = Y$ и такая же подпоследовательность символов-разделителей $T_Q^{m_n^{s,p}} = T_Y^m$.

Шаг 4. Если для шаблона $\left(\left(w_q\right)_1^Q\right)^{m_n^{s,p}}$ найдется такой $\left(\left(w_y\right)_1^Y\right)^m \in W$, что хотя бы половина их слов совпадает, т. е. $\left(\left(w_q\right)_1^Q\right)^{m_n^{s,p}} \cap \left(\left(w_y\right)_1^Y\right)^m \ge \left(w_q\right)_1^{\left[\frac{Q}{2}\right]}$, то считаем, что $\left(\left(w_q\right)_1^Q\right)^{m_n^{s,p}}$ и $\left(\left(w_y\right)_1^Y\right)^m$ подобны (соответствуют одному и тому же типу ЖС, см. рис. 2). Тогда элементы w, которые составляют разницу между $\left(\left(w_q\right)_1^Q\right)^{m_n^{s,p}}$ и $\left(\left(w_y\right)_1^Y\right)^m$, определяем как поля-характеристики $\{f_1, ..., f_J\}$, где J — максимально возможное количество полей для $\left(\left(w_q\right)_1^Q\right)^m$. Перейти к шагу 6.



■ ***Рис. 3.*** Алгоритм построения регулярных выражений

■ ***Fig. 3.*** Flowchart of the regular expression construction algorithm

Шаг 5. Если для шаблона $\left(\left(w_q\right)_1^Q\right)^{m_n^{s,p}}$ не найдется такой $\left(\left(w_y\right)_1^Y\right)^m \in W$, что хотя бы половина их слов совпадает, т. е. $\left(\left(w_q\right)_1^Q\right)^{m_n^{s,p}} \cap \left(\left(w_y\right)_1^Y\right)^m \geq \geq \left(w_q\right)_1^{\left\lceil \frac{Q}{2} \right\rceil}$, то считаем, что $\left(\left(w_q\right)_1^Q\right)^{m_n^{s,p}}$ – новый уникальный шаблон (новый тип ЖС). Завершить итерацию алгоритма.

Шаг 6. Для $\left(\left(w_q\right)_1^Q\right)^{m_n^{s,p}}$ выполняется преобразование $\mathrm{Re}\, g\left(\left(\left(w_q\right)_1^Q\right)^{m_n^{s,p}}, T_Q^{m_n^{s,p}}\right) = r_h$, $h \in [1; H]$, $r_h \in R$, где H – количество полученных шаблонов; R – множество регулярных выражений [2]. В рамках преобразования слова, выделенные на шаге 4 как поля, заменяются на «.*» и выделяются в группы РВ r_h. Группы считаются, начиная с 1-й, группа 0 – все сообщение целиком. Завершить итерацию алгоритма.

Представленный алгоритм можно изобразить в виде блок-схемы (рис. 3).

Оценка временной сложности алгоритма построения РВ: $O(q) + O(k) << \infty$, где q – количество слов в шаблоне ЖС, а k – количество старых шаблонов, которые удовлетворяют условиям $Q = Y$ и $T_Q^{m_n^{s,p}} = T_Y^m$.

Реализация системы построения регулярных выражений

Для эффективной работы алгоритма построения РВ необходимо реализовать систему хранения накопленных шаблонов ЖС и соответствующих им РВ. Для решения этой проблемы была спроектирована база данных (БД) SQlite3 (https://www.sqlite.org/). БД представляется в виде двух таблиц:

– templates – таблица для хранения шаблонов:

- id – идентификатор записи шаблона;
- template – строка для хранения шаблона ЖС;

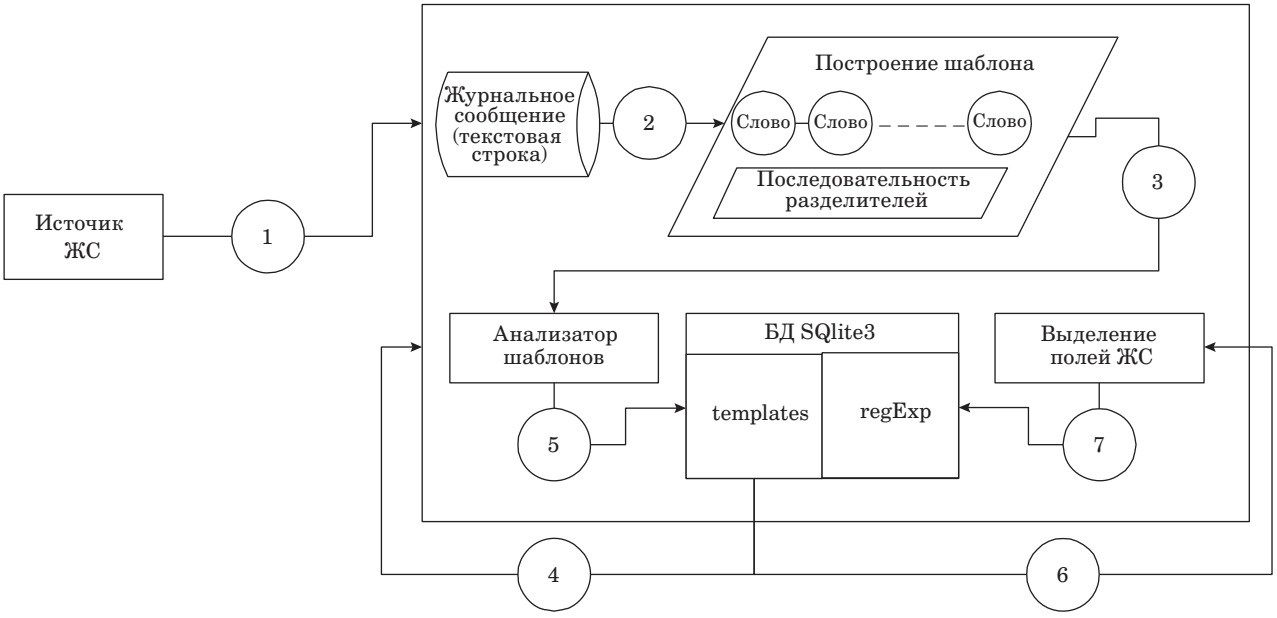
– regExp – таблица для хранения РВ:

- id – целочисленный идентификатор записи РВ;
- idt – целочисленный идентификатор записи шаблона, которому соответствует запись РВ;
- regular – строка для хранения РВ.

Записи шаблонов и соответствующих им РВ в таблицах templates и regExp связаны через идентификаторы биективно. Если в процессе работы алгоритма построения РВ какой-либо шаблон будет скорректирован в соответствии с шагом 3 алгоритма, то соответствующая ему запись РВ в БД будет также изменена.

Обобщенная схема программной реализации построения РВ ЖС представлена на рис. 4.

1. Журнальное сообщение создается источником и отправляется по одному из протоколов.



■ ***Рис. 4.*** Этапы получения и преобразования записей ЖС в РВ

■ ***Fig. 4.*** Steps involved in the collection and conversion of log entries to regular expressions

2. На основании текстовой строки ЖС создается его шаблон.

3. Шаблон передается в анализатор.

4. Из таблицы templates запрашиваются уже существующие шаблоны такой же длины и с той же последовательностью символов-разделителей.

5. Шаблон ЖС сравнивается с полученными шаблонами из БД. На основании результатов сравнения анализатором принимается решение о приведении шаблонов к общему виду, либо о внесении нового шаблона, либо об отсутствии необходимости в этих действиях.

6. В шаблоне общего вида выделяются поля (для структуризации ЖС).

7. Шаблон общего вида с выделенными полями, характеризующими ЖС (регулярное выражение), вносится в таблицу regExp как новое или измененное (в соответствии с п. 6) РВ.

Обобщенная схема реализована в форме программного комплекса, позволяющего преобразовывать ЖС в РВ.

Экспериментальные исследования

Апробация алгоритма построения РВ производилась как на физических, так и на виртуальных устройствах, которые были включены в реальные ВС различных конфигураций

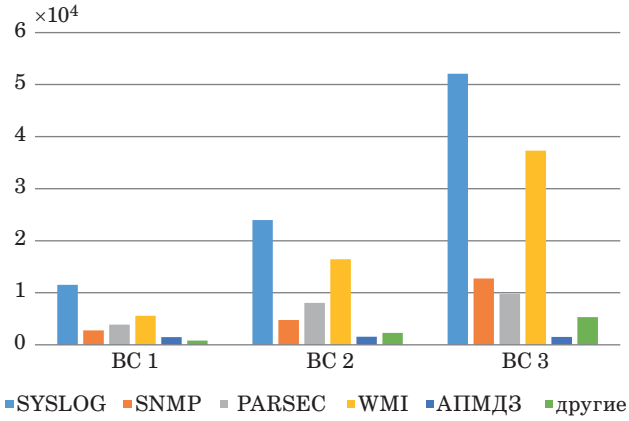
ВС 1. Локальная сеть с выходом в сеть Интернет, включающая: 15 компьютеров под управлением ОС Astra linux 1.5, семь компьютеров под управлением ОС Astra linux 1.6, 10 компьютеров под управлением ОС Astra linux 8.1, 12 компьютеров под управлением ОС Windows 10, два компьютера под управлением ОС Windows Server 2012, три устройства Cisco 2960, два устройства HUAWEI WiFi AX2, три устройства Dionis NX, два устройства Canon MF461DW.

ВС 2. Локальная сеть без выхода в сеть Интернет, включающая: пять компьютеров под управлением ОС Alt linux, 18 компьютеров под управлением ОС Astra linux 1.6, 27 компьютеров под управлением ОС Astra linux 8.1, 12 компьютеров под управлением ОС Windows 10, 14 компьютеров под управлением ОС Windows 11, 12 компьютеров под управлением ОС Windows 8.1, один компьютер под управлением ОС CentOS linux, пять устройств Cisco 2951, три устройства Juniper MX Series, одно устройство MikroTik CRS504-4XQ-IN, одно устройство RUSTELETEH RTT-M300.

ВС 3. Локальная сеть с выходом в сеть Интернет, включающая: 35 компьютеров под управлением ОС Astra linux 1.6, 27 компьютеров под управлением ОС Astra linux 8.1, 54 компьютера под управлением ОС Windows 10, 43 компьютера под управлением ОС Windows 11, 41 компьютер под управлением ОС CentOS linux, пять компьютеров под управлением ОС Ubuntu linux, два устройства Canon MF461DW, 23 устройства D-link DSA-2208X, восемь устройств MikroTik CRS504-4XQ-IN, два устройства «Поток КМ-122», 15 межсетевых экранов ССПТ2.

Вычислительные системы, построенные для проведения испытаний, продуцировали ЖС разных форматов (рис. 5). Оценка скорости появления новых ЖС показала неравномерные результаты, зависящие от различных факторов: количества аппаратных и программных компонент, составляющих ВС, их загруженности, установленных конфигураций журналирования, количества ошибок в ВС и других внешних факторов (например, обработки запросов от клиентов на интернет-странице). В зависимости от изложенных аспектов скорость наполнения журналов сообщениями может варьироваться. Испытания алгоритма построения РВ производились в течение 168 ч непрерывного эксперимента для каждой ВС во время их эксплуатации. За этот период минимальный объем получаемых сообщений был зафиксирован на уровне 2352 сообщений в секунду, а максимальный — 739 681 сообщения в секунду. В среднем же представленные ВС продуцировали 60 000–70 000 сообщений в секунду.

Можно заметить (см. рис. 5), что основной объем ЖС во всех ВС сгенерирован в формате SYSLOG. Это объясняется тем, что SYSLOG — базовая система журналирования для всех UNIX подобных операционных систем. Благодаря этому программное обеспечение, написанное для linux-систем, чаще всего использует возможности, предоставляемые операционной системой (SYSLOG) для ведения журналов. Кроме того,



■ ***Рис. 5.*** Распределение журнальных сообщений по протоколам доставки в наблюдаемых системах, продуцируемых в среднем за секунду

■ ***Fig. 5.*** Log message distribution by transport protocols across monitored systems (average per-second generation rate)

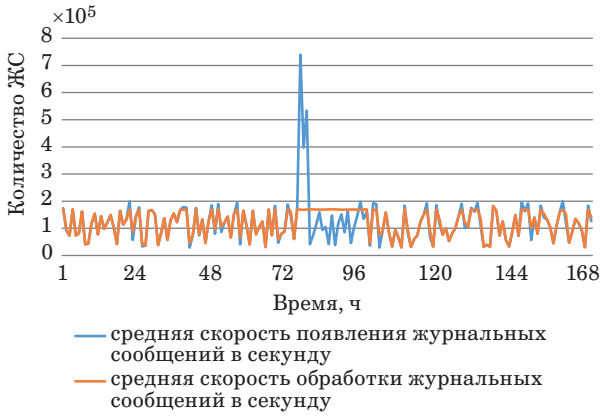
в телекоммуникационном оборудовании чаще всего используются UNIX подобные операционные системы, что также сказывается на количестве ЖС, генерируемых ВС в этом формате. Протокол SNMP используется намного реже, чем SYSLOG (см. рис. 5). Несмотря на то, что протокол SNMP можно настроить как для ОС linux, так и ОС Windows, чаще всего он используется только в телекоммуникационном оборудовании, где настроен сразу на базовом уровне, ввиду удобства использования возможности активных GET-запросов для опроса состояния сетевого оборудования. Система журналирования PARSEC также не отличается большим количеством продуцируемых экземпляров сообщений, так как является частью одноименной системы разграничения прав доступа и генерирует сообщения, отражающие только важные события, связанные с ней. Протокол WMI [25] является внутренним решением для ОС Windows. Количество сообщений, создаваемых в этом формате в среднем, пропорционально количеству устройств, использующих ОС Windows в ВС. Количество сообщений АПМДЗ в целом остается на одном уровне для всех ВС, так как зависит от количества используемых физических плат (которых было в недостаточном количестве — всего по две платы в каждой ВС). Другие ЖС в ВС продуцировались отдельными экземплярами программного обеспечения, которые не использовали штатные средства операционных систем для ведения журналов.

Первоначально программная реализация алгоритма построения РВ тестировалась на компьютере на базе процессора «Эльбрус 8СВ» (архитектура e2k, восемь физических ядер (http://mcst.ru/Elbrus-8CB)), под управлением ОС Astra linux 8.1. Данный компьютер присутствовал во всех тестируемых ВС. Обработка ЖС осуществлялась со скоростью 3700–4000 сообщений в секунду. Этого было недостаточно для обработки всего объема ЖС в режиме реального времени. Поэтому был применен метод вертикального масштабирования. Блок обработки информации в алгоритме построения РВ был распараллелен на этапе выделения шаблонов на шесть потоков. Одно ядро процессора отводилось под сведение результатов выделения шаблонов и сравнение с другими шаблонами из БД. Еще одно ядро было зарезервировано для операционной системы, чтобы не потерять управление в случае чрезмерной нагрузки. В таком режиме удалось достичь ощутимой прибавки в производительности: скорость обработки сообщений возросла до 19 000–20 000 в секунду. Полученной производительности оказалось достаточно для обработки сообщений ВС 1 в реальном времени. Однако для обработки журналов ВС 2 и ВС 3 пришлось применить метод горизонтального масштабирования (разделение обработки ЖС между тремя компьютерами для ВС 2 и шестью компьютерами для ВС 3 на базе процессора «Эльбрус 8СВ»).

Были проведены дополнительные исследования на компьютере на базе процессора Intel Xeon Platinum 8362 (архитектура x86, 32 физических ядра, 64 логических ядра (https://www.intel.com/content/www/us/en/products/sku/217216/intel-xeon-platinum-8362-processor-48m-cache-2-80-ghz/specifications.html)), под управлением ОС Astra linux 1.6. На тех же шести потоках получилось достичь скорости 25 000–27 000 сообщений в секунду. При увеличении количества потоков выделения шаблонов до 20 штук прирост производительности наблюдался близким к линейному и составил 83 000–85 000 сообщений в секунду, однако затем, с увеличением потоков обработки, прирост производительности начал замедляться из-за ограничения производительности процессора на одно ядро в задаче сведения выделенных шаблонов и их сравнения. Если производительность алгоритма не удовлетворяет требованиям реального времени, то необходимо использовать либо процессор с большей производительностью на ядро (при том же количестве ядер), либо метод горизонтального масштабирования.

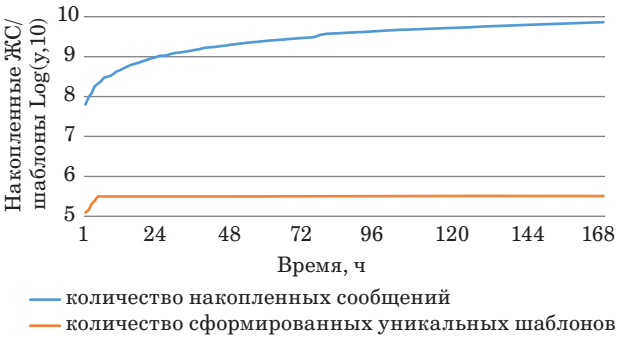
В целом все системы (ВС 1, ВС 2 и ВС 3) похожи в плане форматов ЖС и отличаются в количестве продуцируемых ЖС. Для ВС 3, генерирующей наибольший объем ЖС, из тестируемых ВС были проведены дополнительные исследования по обработке ЖС алгоритмом. ЖС обрабатывались на двух компьютерах под управлением ОС Astra linux 1.6, построенных на процессорах Intel Xeon Platinum 8362.

На графике (рис. 6) отображено сравнение скоростей разрастания журналов (среднее количество ЖС, генерируемых за секунду в течение



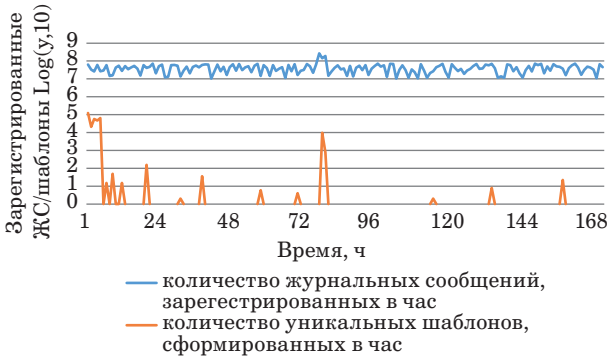
■ ***Рис. 6.*** Результаты апробации алгоритма генерации регулярных выражений в контуре ВС 3

■ ***Fig. 6.*** Validation results of the regex generation algorithm in computing system 3's environment



■ ***Рис. 7.*** Влияние роста журналов на накопление новых шаблонов в ВС 3

■ ***Fig. 7.*** Correlation between log size growth and pattern discovery rate in computing system 3



■ ***Рис. 8.*** Влияние роста журналов (журнальных сообщений) на количество сформированных уникальных шаблонов в час в ВС 3

■ ***Fig. 8.*** Effect of increasing log volume on hourly unique pattern generation rate in computing system 3

■ ***Таблица 2.*** Примеры структуризации ЖС

■ ***Table 2.*** Sample structured log message representations

Протокол	Журнальное сообщение	Регулярное выражение	Выделенные поля
SYSLOG	Dec 3 19:17:01 a161 CRON[25552]: pam_unix(cron:session): session opened for user dr1 by (uid = 1)	(.*) a161 CRON\[(.*)\]\: pam_unix\(cron\:session\)\: session opened for user (.*) by \(uid = (.*)\)	Dec 3 19:17:01 25552 dr1 1
PARSEC	[f] 'Tue Feb 18 06:53:59 2025' '/bin/atsrv' <21588,12096,0,0,0> [s] open("/usr/argusids/lib64/libtransp.so",NO_PERMS | O_LARGEFILE) = 0	\[f\] \'(.*)\' \'(.*)\' \<(.*)\,(.*)\,0\,0\,0\> \[s\] open\(\"(.*)\"\,NO_PERMS | O_LARGEFILE\) = 0	Tue Feb 18 06:53:59 2025 /bin/atsrv 21588 12096 /usr/argusids/lib64/libtransp.so
АПМДЗ	54;Log entry:p_support;User:218f0b8a-834d-0e74-b42c-aca4636cda97;Type:Событие НСД;Code:LOG_CRC_MISMATCH_FILE;File:$MountPoint/1/home/user/apmdz/log/ahsm_log;Severity:LOG_SEVERITY_ALERT;Log DateTime: Mon Feb 17 10:06:12 2025	(.*);Log entry:(.*);User:(.*);Type:Событие НСД;Code:(.*);File:(.*);Severity:(.*);Log DateTime: (.*)	54 p_support 218f0b8a-834d-0e74-b42c-aca4636cda97 LOG_CRC_MISMATCH_FILE $MountPoint/1/home/user/apmdz/log/ahsm_log LOG_SEVERITY_ALERT Mon Feb 17 10:06:12 2025
SNMP	1_3_6_1_4_1_9_9_41_1_2_3_1_2_3="LINK";1_3_6_1_4_1_9_9_41_1_2_3_1_3_3=4;1_3_6_1_4_1_9_9_41_1_2_3_1_4_3="UPDOWN";1_3_6_1_4_1_9_9_41_1_2_3_1_5_3="Interface FastEthernet0/2, changed state to up";1_3_6_1_4_1_9_9_41_1_2_3_1_6_3=(182210813)	1_3_6_1_4_1_9_9_41_1_2_3_1_2_3="LINK";1_3_6_1_4_1_9_9_41_1_2_3_1_3_3=(.*);1_3_6_1_4_1_9_9_41_1_2_3_1_4_3="UPDOWN";1_3_6_1_4_1_9_9_41_1_2_3_1_5_3="(.*)";1_3_6_1_4_1_9_9_41_1_2_3_1_6_3=(.*)	4 Interface FastEthernet0/2, changed state to up (182210813)
WMI	Сведения 27.02.2025 12:58:56 Microsoft-Windows-Kernel-Boot 25 (32) «Использовалась следующая политика меню загрузки: 0x1.»	Сведения (.*)Microsoft-Windows-Kernel-Boot 25 (.*) \"(.*)\"	27.02.2025 12:58:56 (32) Использовалась следующая политика меню загрузки: 0x1.

одного часа) и обработки ЖС алгоритмом генерации РВ. Для моделирования внештатной ситуации на 77-м часе эксперимента была произведена искусственная DdoS-атака [26] на один из компьютеров ВС. В результате произошел наблюдаемый всплеск регистрируемых ЖС — 739 681. Как видно из результатов эксперимента (рис. 6–8), алгоритм построения РВ справился с обработкой многократно возросшего количества ЖС в реальном времени. Также в рамках эксперимента в ВС 3 периодически добавлялись отдельные устройства и компьютеры, однако они не оказали видимого влияния на результаты.

На графике (см. рис. 7) показан постоянный рост количества зарегистрированных в ВС ЖС, при этом уникальные шаблоны практически перестают накапливаться после пятого часа. Влияния DdoS-атаки с 77-го по 79-й час на формирование уникальных шаблонов не наблюдается. Это объясняется тем, что хотя ВС продуцирует большое количество ЖС во время атаки, они однотипны, поэтому уникальных шаблонов создается немного на фоне общего их числа.

Данные на графике (см. рис. 8) для показательности прологарифмированы. Точки, где нет новых шаблонов, обнаруженных в течение часа, выколоты. Небольшие всплески между накоплением шаблонов в начале и DdoS-атакой, а также после нее объясняются регистрацией редких ЖС и добавлением в ВС 3 отдельных устройств в ходе эксперимента.

Дальнейшие исследования показали, что наращивание ресурсов ВС не накладывает ограничений на обработку ЖС в реальном масштабе времени за счет распределенных вычислений.

Примеры структуризации ЖС различных форматов при помощи РВ, построенных в результате апробации разработанного алгоритма для ВС 1, ВС 2 и ВС 3, представлены в табл. 2.

В результате апробации алгоритма в контурах ВС 1, ВС 2 и ВС 3 все генерируемые ЖС были обработаны без потерь, вне зависимости от конфигурации источников, формата и частоты появления. Долгосрочных временных задержек при обработке не наблюдалось.

Заключение

В отличие от существующих научных решений [2–4, 6–24], а также коммерческих решений (см. табл. 1), предложенный алгоритм построения регулярных выражений для журнальных сообщений произвольных вычислительных систем позволяет структурировать ЖС (выделять характеристики) в реальном масштабе времени независимо от их (ЖС) формата, набора характеристик и частоты встречаемости в ВС.

Проведенные исследования алгоритма на реальных ВС показали, что вертикальное масштабирование, горизонтальное масштабирование или их комбинация позволяют обрабатывать ЖС в реальном времени для ВС любой конфигурации.

Структурированные данные ЖС могут быть использованы в анализаторах аномалий и компьютерных атак в ВС, актуальны для расследований инцидентов информационной безопасности и для аудита ВС.

Литература

1. **Zhu J., He S., Liu J., He P., Xie Q., Zheng Z., Lyu M. R.** Tools and benchmarks for automated log parsing. *Proceedings — 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice, ICSE-SEI*, 2019. doi:10.48550/arXiv.1811.03509
2. **Драчев Г. А.** Разработка алгоритма выделения и кодирования данных из журнальных сообщений вычислительной системы для систем обнаружения аномалий. *Информационные технологии*, 2023, т. 29, № 7, с. 351–359. doi:10.17587/it.29.351-359
3. **Савицкий Д. Е., Дунаев М. Е., Зайцев К. С.** Выявление аномалий при обработке потоковых данных в реальном времени. *International Journal of Open Information Technologies*, 2022, т. 10, № 6, с. 70–76.
4. **Du M., Li F., Zheng G., Srikumar V.** Deeplog: Anomaly detection and diagnosis from system logs through deep learning. *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 1285–1298. doi:10.1145/3133956.3134015
5. **Shendi M. M., Elkadi H. M., Khafagy M. H.** A study on the big data log analysis: goals, challenges, issues, and tools. *International Journal of Soft Computing and Artificial Intelligence*, 2019, vol. 7, pp. 1–12.
6. **Абрамов А. Г.** Высокопроизводительный сервис сбора и анализа файлов журналов сетевого и серверного оборудования в национальной исследовательской компьютерной сети. *Программные продукты и системы*, 2024, т. 37, № 4, с. 495–503. doi:10.15827/0236-235X.148.495-503
7. **Хасанова А. М.** Интеллектуальный анализ процессов по данным журналов событий информационных систем. *International Journal of Open Information Technologies*, 2022, т. 10, № 10, с. 70–77.
8. **Костиков Е. В.** Методы анализа логов Sysmon для обнаружения киберугроз. *International Journal of Open Information Technologies*, 2024, т. 12, № 11, с. 25–34.
9. **Гумеров Б. З.** Методы обогащения событий информационной безопасности с помощью CRIBL и

MISP. *Проблемы современной науки и образования*, 2022, № 6 (175), с. 38–45.
10. **Li Z., Fu Q., Huang Z., Yu Y., Lai Y., Ma Y.** Revisiting log parsing: The present, the future, and the uncertainties. *IEEE Transactions on Reliability*, 2024, pp. 1–14. doi:10.1109/TR.2023.3340020
11. **Chen S., Liao H.** BERT-log: Anomaly detection for system logs based on pre-trained language model. *Applied Artificial Intelligence*, 2022, vol. 36, no. 1, pp. e21456422028014. doi:10.1080/08839514.2022.2028014
12. **Hu J., Long L., Sui H., Gu Z., Zheng G.** CFTL: System log parsing method driven from clustering according to first token and length for anomaly detection. *Applied Sciences*, 2025, vol. 15, pp. 1740. doi:10.3390/app15041740
13. **Jin Bo Chen, Wen Yu Hu, Kang Hui Ying, Guo Nong Li.** A log analysis technology based on FP-growth improved algorithm. *2021 International Conference on Artificial Intelligence. Big Data and Algorithms (CAIBDA*), Xi'an, 2021, 28–30 May, 2021, pp. 219–223. doi:10.1109/CAIBDA53561.2021.00053
14. **Xiao T., Quan Z., Wang Z., Zhao K., Liao X., Huang H., Du Y., Li K.** LPV: A log parsing framework based on vectorization. *IEEE Transactions on Network and Service Management*, 2023, pp. 1. doi:10.1109/TNSM.2023.3248124
15. **Chen X., Wang P., Chen J., Wang W.** AS-parser: Log parsing based on adaptive segmentation. *Proceedings of the ACM on Management of Data*, 2023, vol. 1, pp. 1–26. doi:10.1145/3626719
16. **Wei M., Wen J., He S., Xie K., Liang W., Xie G., Li K., Zhu Z.** TCMS: A multi-sequence log parsing method based on token conversion. *IEEE Transactions on Dependable and Secure Computing*, 2024, pp. 1–18. doi:10.1109/TDSC.2024.3520628
17. **Yu S., He P., Chen N., Wu Y.** Brain: Log parsing with bidirectional parallel tree. *IEEE Transactions on Services Computing*, 2023, pp. 1–12. doi:10.1109/TSC.2023.3270566
18. **Чаругин В. В., Чаругин В. В., Чесалин А. Н., Ушкова Н. Н.** Конструктор блоков обработки естественного языка и применение его в задаче структурирования логов в информационной безопасности. *International Journal of Open Information Technologies*, 2024, т. 12, № 9, с. 111–119.
19. **Liu Y., Wu Y., Song W., Chen Z., Li Z.** LogPrompt: Prompt engineering towards zero-shot and interpretable log analysis. *Proceedings of the 2024 IEEE/ACM 46th International Conference on Software Engineering: Companion Proceedings (ICSE)*, 2024, pp. 364–365. doi:10.1145/3639478.3643108
20. **Zhang C., Xu W., Liu J., Zhang L., Liu G., Guan J., Zhou Q., Zhou S.** LogBase: A large-scale benchmark for semantic log parsing. *Proceedings of the ACM on Software Engineering*, 2025, vol. 2, pp. 2091–2112. doi:10.1145/3728969
21. **Jiang Z., Liu J., Chen Z., Li Y., Huang J., Huo Y., He P., Gu J., Lyu M.** LILAC: Log parsing using LLMs with adaptive parsing cache. *Proceedings of the ACM on Software Engineering*, 2024, vol. 1, pp. 137–160. doi:10.1145/3643733
22. **Liu S., Yun L., Nie S., Zhang G., Li W.** IPLog: An efficient log parsing method based on few-shot learning. *Electronics*, 2024, vol. 13, pp. 3324. doi:10.3390/electronics13163324
23. **Cheng H., Ying S., Duan X., Yuan W.** DLLog: An online log parsing approach for large-scale system. *International Journal of Intelligent Systems*, 2024, pp. 1–17. doi:10.1155/2024/5961993
24. **Rücker N., Maier A.** FlexParser—The adaptive log file parser for continuous results in a changing world. *Journal of Software: Evolution and Process*, 2022, pp. 34. doi:10.1002/smr.2426
25. **Маркин Д. И., Гортинский А. В.** Использование технологии WMI для сбора информации и отслеживания событий в ОС Windows. *Информационная безопасность регионов*, 2016, № 4 (25), с. 11–15.
26. **Jun J.-H., Oh H., Kim S.-H.** DDoS flooding attack detection through a step-by-step investigation. *2011 IEEE 2nd International Conference on Networked Embedded Systems for Enterprise Applications*, Perth, 2011, 8–9 December, 2011, pp. 1–5. doi:10.1109/NESEA.2011.6144944

UDC 004.492.3
doi:10.31799/1684-8853-2026-1-36-47
EDN: LSYLOP

Algorithm for automatic construction of regular expressions for preprocessing arbitrary-format log messages in computing systems

G. A. Drachev[a,b], Researcher, Visiting Lecturer, Project Manager, orcid.org/0000-0003-1851-5507, pendal2@gmail.com
[a]National Research University Higher School of Economics, 34, Tallinskaya St. 123458, Moscow, Russian Federation
[b]Special System Engineering Center — Service, 71, Varshavskoe Highway, 117556, Moscow, Russian Federation

Introduction: Preprocessing is necessary to structure log messages by extracting characteristic fields for subsequent analysis aimed at detecting anomalies and cyber attacks in the computing system. The most acceptable approach, which dominates commercial solutions, is the construction of regular expressions corresponding to log messages. However, creating these regular expressions requires labor-

intensive manual processing. **Purpose:** To develop an algorithm for the automatic construction of regular expressions for log messages in real time, applicable to any log message regardless of its format or source. **Results:** The analysis of log message sources, their delivery methods to storage and processing systems, and existing log message formats has shown that log messages, even within a single format, often lack standardization in field composition. We have developed an algorithm to generate a template corresponding to a given log message and, through template processing, construct a regular expression with extracted fields effectively structuring the log message. Additionally, we have designed a system for storing accumulated log message templates and their corresponding regular expressions. For experimental research, we have developed a software toolkit to automatically construct regular expressions from log message texts. The toolkit has been tested on real computing systems with various configurations. **Practical relevance:** The proposed algorithm enables structuring of log messages of arbitrary types and formats. Structured log messages can be used for cybersecurity incident investigations, information system audits, and as input data for anomaly and cyberattack analyzers.

Keywords — information security, system logs, log messages, data preprocessing, log message structuring, regular expression.

For citation: Drachev G. A. Algorithm for automatic construction of regular expressions for preprocessing arbitrary-format log messages in computing systems. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2026, no. 1, pp. 36–47 (In Russian). doi:10.31799/1684-8853-2026-1-36-47, EDN: LSYLOP

References

1. Zhu J., He S., Liu J., He P., Xie Q., Zheng Z., Lyu M. R. Tools and benchmarks for automated log parsing. *Proceedings — 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice, ICSE-SEI*, 2019. doi:10.48550/arXiv.1811.03509
2. Drachev G. A. Development of an algorithm for extracting and encoding data from log messages of a computing system for anomaly detection systems. *Information Technologies*, 2023, vol. 29, no. 7, pp. 351–359 (In Russian). doi:10.17587/it.29.351-359
3. Savitsky D. E., Dunaev M. E., Zaytsev K. S. Anomaly detection in real-time streaming data processing. *International Journal of Open Information Technologies*, 2022, vol. 10, no. 6, pp. 70–76 (In Russian).
4. Du M., Li F., Zheng G., Srikumar V. Deeplog: Anomaly detection and diagnosis from system logs through deep learning. *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 1285–1298. doi:10.1145/3133956.3134015
5. Shendi M. M., Elkadi H. M., Khafagy M. H. A study on the big data log analysis: goals, challenges, issues, and tools. *International Journal of Soft Computing and Artificial Intelligence*, 2019, vol. 7, pp. 1–12.
6. Abramov A. G. High-performance service for collecting and analyzing network and server hardware log files on a National Research Computer Network. *Software & Systems*, 2024, vol. 37, no. 4, pp. 495–503 (In Russian). doi:10.15827/0236-235X.148.495-503
7. Khasanova A. M. Process mining methods to analyze event logs of information systems. *International Journal of Open Information Technologies*, 2022, vol. 10, no. 10, pp. 70–77 (In Russian).
8. Kostikov E. V. Sysmon log analysis methods for cyber threat detection. *International Journal of Open Information Technologies*, 2024, vol. 12, no. 11, pp. 25–34 (In Russian).
9. Gumerov B. Z. Methods for the enrichment of infomation security events using CRIBL and MISP. *Problemy sovremennoj nauki i obrazovaniya*, 2022, no. 6 (175), pp. 38–45 (In Russian).
10. Li Z., Fu Q., Huang Z., Yu Y., Lai Y., Ma Y. Revisiting log parsing: The present, the future, and the uncertainties. *IEEE Transactions on Reliability*, 2024, pp. 1–14. doi:10.1109/TR.2023.3340020
11. Chen S., Liao H. BERT-log: Anomaly detection for system logs based on pre-trained language model. *Applied Artificial Intelligence*, 2022, vol. 36, no 1, pp. e21456422028014. doi:10.1080/08839514.2022.2028014
12. Hu J., Long L., Sui H., Gu Z., Zheng G. CFTL: System log parsing method driven from clustering according to first token and length for anomaly detection. *Applied Sciences*, 2025, vol. 15, pp. 1740. doi:10.3390/app15041740
13. Jin Bo Chen, Wen Yu Hu, Kang Hui Ying, Guo Nong Li. A log analysis technology based on FP-growth improved algorithm. *2021 International Conference on Artificial Intelligence. Big Data and Algorithms (CAIBDA)*, Xi'an, 2021, pp. 219–223. doi:10.1109/CAIBDA53561.2021.00053
14. Xiao T., Quan Z., Wang Z., Zhao K., Liao X., Huang H., Du Y., Li K. LPV: A log parsing framework based on vectorization. *IEEE Transactions on Network and Service Management*, 2023, pp. 1. doi:10.1109/TNSM.2023.3248124
15. Chen X., Wang P., Chen J., Wang W. AS-parser: Log parsing based on adaptive segmentation. *Proceedings of the ACM on Management of Data*, 2023, vol. 1, pp. 1–26. doi:10.1145/3626719
16. Wei M., Wen J., He S., Xie K., Liang W., Xie G., Li K., Zhu Z. TCMS: A multi-sequence log parsing method based on token conversion. *IEEE Transactions on Dependable and Secure Computing*, 2024, pp. 1–18. doi:10.1109/TDSC.2024.3520628
17. Yu S., He P., Chen N., Wu Y. Brain: Log parsing with bidirectional parallel tree. *IEEE Transactions on Services Computing*, 2023, pp. 1–12. doi:10.1109/TSC.2023.3270566
18. Charugin V. V., Charugin V. V., Chesalin A. N., Ushkova N. N. Constructor of natural language processing blocks and its application in the problem of structuring logs in information security. *International Journal of Open Information Technologies*, 2024, vol. 12, no. 9, pp. 111–119 (In Russian).
19. Liu Y., Wu Y., Song W., Chen Z., Li Z. LogPrompt: Prompt engineering towards zero-shot and interpretable log analysis. *Proceedings of the 2024 IEEE/ACM 46th International Conference on Software Engineering: Companion Proceedings (ICSE)*, 2024, pp. 364–365. doi:10.1145/3639478.3643108
20. Zhang C., Xu W., Liu J., Zhang L., Liu G., Guan J., Zhou Q., Zhou S. LogBase: A large-scale benchmark for semantic log parsing. *Proceedings of the ACM on Software Engineering*, 2025, vol. 2, pp. 2091–2112. doi:10.1145/3728969
21. Jiang Z., Liu J., Chen Z., Li Y., Huang J., Huo Y., He P., Gu J., Lyu M. LILAC: Log parsing using LLMs with adaptive parsing cache. *Proceedings of the ACM on Software Engineering*, 2024, vol. 1, pp. 137–160. doi:10.1145/3643733
22. Liu S., Yun L., Nie S., Zhang G., Li W. IPLog: An efficient log parsing method based on few-shot learning. *Electronics*, 2024, vol. 13, pp. 3324. doi:10.3390/electronics13163324
23. Cheng H., Ying S., Duan X., Yuan W. DLLog: An online log parsing approach for large-scale system. *International Journal of Intelligent Systems*, 2024, pp. 1–17. doi:10.1155/2024/5961993
24. Rücker N., Maier A. FlexParser—The adaptive log file parser for continuous results in a changing world. *Journal of Software: Evolution and Process*, 2022, pp. 34. doi:10.1002/smr.2426
25. Markin D. I., Gortinsky A. V. Using WMI technology to collect data and trace events in Windows operating system. *Informacionnaja bezopasnost' regionov*, 2016, no. 4(25), pp. 11–15 (In Russian).
26. Jun J.-H., Oh H., Kim S.-H. DDoS flooding attack detection through a step-by-step investigation. *2011 IEEE 2nd International Conference on Networked Embedded Systems for Enterprise Applications*, Perth, 2011, pp. 1–5. doi:10.1109/NESEA.2011.6144944

УДК 004.056:004.75
doi:10.31799/1684-8853-2026-1-48-60
EDN: AVYJSY

Научные статьи
Articles

Использование графов для детектирования бесфайловых атак в контейнеризированной инфраструктуре

Е. О. Здорников[а], *инженер-программист, orcid.org/0009-0009-0154-5153*
Е. В. Егоров[б], *младший научный сотрудник, orcid.org/0009-0006-4321-0069*
И. Ю. Попов[а], *канд. техн. наук, доцент, orcid.org/0000-0002-6407-7934, ilyapopov27@gmail.com*
[а]*Университет ИТМО, Кронверкский пр., 49, Санкт-Петербург, 197101, РФ*
[б]*Военно-космическая академия им. А. Ф. Можайского, Ждановская наб., 13, Санкт-Петербург, 197198, РФ*

Введение: *бесфайловые атаки позволяют использовать память для исполнения вредоносного кода без сохранения на диск. Это существенно усложняет их обнаружение и делает традиционные методы защиты неэффективными, особенно в динамических контейнерных средах. Контейнерные системы, например Docker и Kubernetes, по сравнению с виртуальными машинами имеют меньший объем контролируемых данных, что упрощает анализ событий и построение графов активности.* ***Цель:*** *разработать метод обнаружения бесфайловых атак в контейнеризированной инфраструктуре, устойчивый к шуму и эвазивным техникам при неполном мониторинге, обеспечивающий раннюю сигнализацию до достижения критических ресурсов.* ***Результаты:*** *предложен риск-центричный метод, который основан на гетерогенном графе и графе системных вызовов. Выделяются зоны риска вокруг событий, зафиксированных с помощью расширенного фильтра пакетов eBPF, после чего формируется математическая модель зон риска с охраняемым замыканием и вычислением потенциала риска на основе поглощающих случайных блужданий. В модели дополнительно учитываются контексты контейнеров и временные параметры, что дает возможность уменьшить количество ложноположительных срабатываний. Метод позволяет локализовать зоны риска и стабильно работает при потере части событий. Эксперименты проводились на кластере Kubernetes (v1.32) под управлением Ubuntu 24.04 с использованием Tetragon (eBPF) и Falco для контроля качества. Собрано 580 эпизодов поведения контейнеров, включая атакующие и фоновые сценарии, на основе которых формировались гетерогенные графы исполнения и системных вызовов. Предложенный метод превосходит статические правила, n-gram-модели системных вызовов и глобальные графовые методы по метрикам AUROC и AUPRC, демонстрируя повышенную устойчивость к эвазивным техникам.* ***Практическая значимость:*** *разработанный метод подходит для работы с существующими сенсорами и политиками контейнерной безопасности и позволяет администраторам получать интерпретируемые зоны риска и показатели вероятности атаки.* ***Обсуждение:*** *представленная математическая модель и практическая реализация подтверждают применимость риск-центричного анализа для практического обнаружения актуальных бесфайловых угроз в современных контейнерных средах; метод масштабируем, параметризуем и не требует модификации приложений.*

Ключевые слова *— бесфайловые атаки, контейнерная безопасность, eBPF, графотемпоральный анализ, runtime-детекция, графы системных вызовов, Kubernetes.*

Для цитирования: Здорников Е. О., Егоров Е. В., Попов И. Ю. Использование графов для детектирования бесфайловых атак в контейнеризированной инфраструктуре. *Информационно-управляющие системы*, 2026, № 1, с. 48–60. doi:10.31799/1684-8853-2026-1-48-60, EDN: AVYJSY

For citation: Zdornikov E. O., Egorov E. V., Popov I. Y. Using graphs to detect fileless attacks in containerized infrastructure. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2026, no. 1, pp. 48–60 (In Russian). doi:10.31799/1684-8853-2026-1-48-60, EDN: AVYJSY

Введение

Бесфайловые (fileless) атаки — это класс угроз, которые преимущественно используют оперативную память для исполнения вредоносного кода без устойчивых артефактов на диске. Современные информационные системы все чаще сталкиваются с бесфайловыми атаками. Такие атаки, в отличие от традиционных методов, не создают файлы на дисковом пространстве и не оставляют очевидных следов. В этом случае вредоносный код хранится и используется только в оперативной памяти, что затрудняет обнаружение антивирусами и системами обнаружения вторжений (Intrusion Detection System — IDS). Согласно обзорной статье [1], атаки без использования файлов становятся все более распространенными среди нарушителей информационной безопасности (ИБ). Например, исследования Ponemon Institute показывают, что вероятность их успешного проведения примерно в десять раз выше по сравнению с традиционными методами атак [1]. Также около 26 % атак APT-группировок (устойчивые целевые атаки, Advanced Persistent Threat) сопряжены с использованием методов, не связанных с файлами, таких как Living-off-the-Land, а с 2022 г. наблюдается рост бесфайловых атак на 70 % [1]. Также исследования Barr-Smith et al показали, что 80 % антивирусов не смогли обнаружить какие-либо соответствующие бесфайловые атаки, в то время как 20 % могли обнаружить только часть атак [1].

Исследование компании WatchGuard Technologies констатировало рост на 888 % бесфайловых угроз в корпоративных сетях за один год, с 2019-го по 2020-й (https://www.watchguard.com/wgrd-resource-center/security-report-q4-2020). Аналогичный отчет от Aqua Nautilus фиксирует увеличение числа таких угроз на 1400 % за 2023 г., что связано с активным применением подобной техники в облачной инфраструктуре [2]. Эти отчеты показывают, что бесфайловые атаки становятся все более популярным и опасным инструментом у нарушителей ИБ. В научных работах последних лет [1, 3, 4] проанализированы вызовы и различные методы для борьбы с такого рода атаками.

Первые бесфайловые вирусы появились в начале 2000-х годов и сразу стали востребованы среди киберпреступников благодаря своей способности действовать без сохранения на диск, тем самым обходя антивирусы. Такие вредоносные программы использовали уязвимости в браузерах, Microsoft Office или скриптовых движках Windows, чтобы загружаться напрямую в память и выполнять вредоносный код, не оставляя следов на файловой системе. Позже злоумышленники стали активно использовать встроенные системные утилиты Windows — так называемые LOLBins и LOLScripts (например, mshta, wscript, certutil, fodhelper, powershell, rundll32 и др.). Эти программы уже подписаны Microsoft, предустановлены в операционной системе и могут использоваться для загрузки файлов, обхода контроля учетных записей пользователей и запуска вредоносных скриптов в обход антивирусов.

Изначально угрозу бесфайловых вредоносных программ связывали только с Windows-системами, но за последние несколько лет они все чаще стали применяться против Linux-серверов и контейнеризированных сред. В Linux также существует аналог LOLBins, включающий в себя предустановленные исполняемые файлы (awk, tar, find, curl, wget, scp, pkexec, nmap, bash, python, perl, less и др.), которые могут быть использованы для выполнения команд, эскалации привилегий, побега из контейнера и организации обратного подключения (reverse shell). Исследования показывают, что нарушители ИБ применяют такие техники, как внедрение кода через системный вызов ptrace, создание исполняемых сегментов памяти с помощью memfd_create() и запуск вредоносных процессов в пространстве /dev/shm, что делает их невидимыми для традиционных средств защиты [5].

В контейнерных инфраструктурах бесфайловые атаки представляют особую опасность, поскольку вредоносный код может выполняться внутри контейнера и оставаться невидимым для средств мониторинга, развернутых на уровне хоста. Контейнерные системы, например Docker и Kubernetes, характеризуются высоким уровнем изоляции на уровне операционной системы, высокой производительностью за счет отсутствия накладных расходов на эмуляцию оборудования и ограниченным доступом к файловой системе. Несмотря на эти преимущества бесфайловые угрозы остаются актуальным вектором атаки для нарушителя ИБ. При этом меньший объем контролируемых данных в контейнерах по сравнению с виртуальными машинами упрощает анализ событий и построения графов активности, что позволяет точнее выделять зоны риска и снижать уровень ложных срабатываний.

В настоящей работе развивается идея риск-центричной детекции: вместо моделирования всего приложения формируются и классифицируются подграфы, что и определяет цель исследования — разработку метода обнаружения бесфайловых атак в контейнерных средах на основе графовых моделей для снижения числа ложных срабатываний и повышения устойчивости детекции при неполном мониторинге.

Предлагаемое решение основано на риск-центричном анализе системных событий контейнера, в котором данные (системные вызовы, сетевые взаимодействия и различные метаданные контейнера), полученные с помощью eBPF (extended Berkeley Packet Filter), преобразуются в гетерогенный граф. На полученном графе выделяются локальные зоны риска, формируется математическая модель взаимосвязей зон риска с охраняемым замыканием и с использованием поглощающих случайных блужданий, что разрешает вычислять потенциал риска для каждого узла и выделять аномальные активности, указывающие на бесфайловую атаку. Такой подход позволяет учитывать контекст контейнера и временную динамику событий, что уменьшает количество ложноположительных срабатываний.

Входными данными для предложенного метода являются телеметрические события, собираемые eBPF-датчиками, включая системные вызовы процессов, сетевые взаимодействия, обращения к файловым дескрипторам, операции с пространствами имен и изменениями привилегий. Эти данные агрегируются в эпизоды наблюдения за поведением контейнера и служат основой для построения гетерогенного графа исполнения.

Актуальность и научная новизна работы заключаются в интеграции риск-центричного подхода с динамическим контекстом контейнеров и временными параметрами событий, что обеспечивает устойчивость к неполному мониторингу, а также снижает уровень ложноположительных срабатываний.

Предложена модель зон риска с охраняемым замыканием для анализа поведения в контей-

нерных средах на основе графов системных вызовов.

Для достижения поставленной цели в работе решаются следующие задачи.

1. Определение модели угроз, учитывающей специфику бесфайловых атак и контейнерных сред.

2. Построение графового представления поведения процессов с классификацией подграфов, пересекающих «зоны риска»:

2.1 разрешения процессов (Linux process capabilities);

2.2 критические пути ввода-вывода и сетевые сокеты;

2.3 переходы между пространствами имен (namespace);

2.4 выполнение кода в памяти (in-memory execution).

3. Формализация математической модели зон риска и правил их пересечения.

4. Экспериментальная проверка.

Экспериментальная проверка показала, что предложенный метод способен стабильно локализовать зоны риска даже при потере части событий. При тестировании на кластере Kubernetes (v1.32) с использованием Tetragon (eBPF) и Falco метод продемонстрировал преимущество по метрикам AUPRC = 0,87 и AUROC = 0,94 по сравнению с сигнатурными правилами Falco/Tetragon, n-gram-моделями системных вызовов и глобальными графовыми подходами без локализации. Отмечено также более медленное падение качества при эвазивных k-модификациях, что указывает на повышенную устойчивость детекции в реальных условиях эксплуатации.

Такой подход снижает поток событий и уровень шума при анализе бесфайловых техник, оставаясь совместимым с eBPF-датчиками и практиками политик контроля во время выполнения (runtime).

Обзор существующих решений

С развитием угроз развивались и средства защиты. Например, в октябре 2018 г. Azure Security Center от компании Microsoft выпустил первое решение для Windows, а уже в 2020-м представил предварительное решение для обнаружения бесфайловых атак на Linux-системах (https://azure.microsoft.com/en-us/blog/fileless-attack-detection-for-linux-in-preview/). Тем не менее эти подходы имеют существенные недостатки. Стандартные IDS и SIEM (управление событиями и информацией безопасности, Security Information and Event Management) малоэффективны против новых бесфайловых угроз. Антивирусные решения в основном используют статический и сигнатурный анализ и не в состоянии детектировать процессы, которые исполняются только в оперативной памяти [5, 6]. IDS-системы, работающие на основе анализа системных вызовов контейнеров, перегружены и довольно часто могут давать большое количество ложных срабатываний, что снижает их практическую ценность [6]. К современным подходам можно отнести следующие решения.

1. Анализ оперативной памяти (memory forensics) позволяет выявить следы вредоносной активности, но требует сложных дампов и ручной обработки [4].

2. Модели на основе системных вызовов (Bag-of-Syscalls) обеспечивают обнаружение аномалий в реальном времени, не требуя предварительных знаний о поведении контейнера [7], однако чувствительны к шуму и могут выдавать большое количество ложных срабатываний при изменении рабочей нагрузки.

3. Использование eBPF-сенсоров (расширенный фильтр пакетов Беркли, extended Berkeley Packet Filter) и kernel-based мониторинга дает возможность отслеживать вызовы на уровне ядра и выявлять нетипичные взаимодействия процессов [8], но требует повышенных привилегий и может оказывать влияние на производительность при высокой интенсивности событий. Эти вызовы могут обрабатываться как статическими правилами, так и методами машинного обучения и нейросетями [9].

4. Графовые модели поведения — один из наиболее перспективных подходов: строятся ориентированные графы системных вызовов, где выделяются подграфы зон риска (например, обращения к docker.sock, попытки изменения пространств имен или доступ к привилегированным файлам). Эти подграфы анализируются с использованием методов машинного обучения и графовых нейронных сетей (Graph Neural Network — GNN), что позволяет обнаруживать сложные и ранее неизвестные паттерны атак [10–12].

Работы последних двух лет показывают эффективность графов зависимостей/происхождения для анализа внутриконтейнерных атак и побегов:

— метод Container Escape Detection [13] на основе графов зависимостей реконструирует причинно-следственные связи и маркирует «контейнерные» процессы на графе, повышая полноту обнаружения escape-сценариев;

— Phoenix [14] строит граф происхождения данных из событий аудита контейнера, что позволяет выявлять «злонамеренные последовательности» и поддерживает динамическую защиту при наличии уязвимых компонентов;

— CORAL [14] использует логические графы атак для онлайн-оценки рисков и выявления ла-

терального перемещения в контейнерных средах.

Данные работы не лишены недостатков. Container Escape Detection ограничен узким сценарием побега из контейнера и требует полного построения графа зависимостей, что повышает чувствительность к шуму и нагрузку на ресурсы. Phoenix ориентирован на заранее известные последовательности системных вызовов и слабо применим к бесфайловым и in-memory атакам. CORAL зависит от полной картины событий и ресурсов контейнера, что снижает эффективность при неполном мониторинге.

Графовое представление поведения системы с выделением зон риска

В настоящей работе модель угроз и способ представления поведения системы рассматриваются совместно, поскольку выбранный способ наблюдения и формализации исполнения напрямую определяет класс обнаруживаемых атак и допустимые допущения безопасности. Основной акцент в работе сделан на исследовании бесфайловых угроз в Linux-контейнерах (Docker/CRI-O, Container Runtime Interface/containerd) под управлением Kubernetes и аналогичных оркестраторов. Фокус — рантайм-поведение, извлекаемое из ядра (eBPF), так как традиционные антивирусы/IDS/SIEM слабо наблюдают in-memory активность и перегружаются шумом, особенно внутри namespaces/cgroups контейнеров. Эти ограничения отмечены и в нашем введении. Предполагается:

1. Доверенная база — ядро хоста и eBPF-сенсор (Falco/Tetragon) не скомпрометированы; доступ к данным телеметрии защищен. На практике именно ядровый контроль исполнения (check-on-execution) показал применимость против in-memory кода до перехода в пространство пользователя (userspace), что важно для бесфайловых сценариев.

2. Наблюдаемость — перехват системных вызовов и LSM-событий (модули безопасности Linux, Linux Security Modules) с контейнерным контекстом (cgroup/kubernetes-labels) возможен и экономичен за счет in-kernel фильтрации/действий (Tetragon Selectors, Falco).

3. Граница нашей работы — анализ рантайм-поведения (включая попытки побега/латерали), но мы не решаем уязвимости цепочек поставок (supply chain) и не рассматриваем компрометацию ядра/сенсора.

Атакующий — удаленный либо внутренний пользователь, получивший начальную точку входа в контейнере: удаленное выполнение кода в приложении, скомпрометированные секреты/образы, ошибочные настройки RBAC (Role-Based Access Control)/Pod-Security, уязвимые скрипты. Привилегии — от непривилегированных до полупривилегированных.

Целями нарушителя могут быть:

1) загрузка и исполнение бесфайловой вредоносной нагрузки;

2) разведка, закрепление, извлечение токенов и внедрение в долгоживущие процессы;

3) попытки выбраться из контейнера (namespace-переходы, mount/pivot_root при наличии CAP_SYS_ADMIN), злоупотребление сокетами рантайма (/var/run/docker.sock, CRI/containerd), доступ к kubelet-артефактам.

После определения модели угроз для бесфайловых атак в контейнерной среде на следующем этапе проводится построение формального графового представления активности системы с помощью графов исполнения, основанных на анализе потока событий ядра, с одновременным выделением зон риска.

1. Поток событий → граф исполнения.

Собирается поток событий ядра (syscalls, LSM-хуки) и преобразуется в графы.

Системные вызовы (syscalls) — интерфейс обращений пользовательских процессов к ядру операционной системы (open, execve, setns, bpf); по ним фиксируются ключевые действия процессов.

LSM-хуки — точки перехвата в ядре, позволяющие контролировать/логировать действия (доступ к файлам, сетям и пр.).

В работе [12] рассматриваются различные графовые модели для анализа вредоносного поведения в системе. Для нашей задачи выявления узких зон риска бесфайловых атак наиболее подходят модели гетерогенного графа (Heterogeneous Graph — HG) и графа системных вызовов (System Call Graph — SCG). Эти модели эффективно отражают ключевые аспекты поведения системы в контейнере: HG моделирует разнообразные объекты и их взаимодействия (процессы, файлы, сокеты, namespaces и т. д.), а SCG фокусируется на последовательности системных вызовов и связях между ними.

Другие рассмотренные модели оказались менее применимыми, так как не учитывают специфику бесфайловых атак или недостаточно полно отражают структуру взаимодействий внутри контейнера.

На основе HG и SCG строим граф исполнения событий, что позволяет выделить узкие зоны риска, характерные для бесфайловых угроз. Такой метод обеспечивает требуемый уровень детализации и полноту анализа, необходимые для эффективного выявления потенциальных атак.

В гетерогенном графе:

— узлы (точки): процессы, файлы, сокеты, namespaces, capability-состояния;

— ребра (линии): взаимодействия между узлами (например, процесс открыл файл, присоединился к namespace);

— метки на ребрах: тип действия, аргументы вызова, время события.

В графе системных вызовов:

— узлы (точки): системные вызовы и процессы;

— ребра (линии): последовательность и связи вызовов (какой syscall вызвал следующий, какие процессы взаимодействуют через syscalls);

— метки на ребрах: тип системного вызова, параметры, временные метки.

Таким образом, формируется графовое представление активности контейнера, в котором можно анализировать взаимодействия и выявлять потенциальные зоны риска. В этом графе сразу выделяем границы — опасные ресурсы и состояния, вроде docker.sock или host namespace.

2. Якоря риска (risk anchors).

Это простые логические события в графе, которые напрямую связаны с известными методами бесфайловых атак. На основе анализа литературы для более четкого представления зон риска в бесфайловых атаках на Linux-контейнеры выделены основные способы, которые чаще всего используют нарушители ИБ.

2.1 Использование временных файловых систем /dev/shm и /run/shm для размещения и исполнения кода в памяти без записи на диск.

2.2 Применение системного вызова memfd_create() для создания анонимных в памяти файлов и исполнения вредоносных бинарников.

2.3 Злоупотребление механизмом LD_PRELOAD для внедрения библиотек и перехвата вызовов функций процесса.

2.4 Использование ptrace для отладки и инъекции кода в другие процессы, обхода защитных механизмов [5].

2.5 Активное применение предустановленных в Linux бинарников из набора GTFObins (например, awk, tar, find, curl, bash) для выполнения команд, установки обратных подключений, эскалации привилегий и побега из контейнеров [15].

2.6 Эксплуатация доступа к сокетам контейнерного рантайма, таким как /var/run/docker.sock, и неправильных namespace-настроек для побега и латерального перемещения [13, 16].

2.7 Извлечение и использование секретов (токенов, ключей) для закрепления и дальнейшего распространения атаки [6].

2.8 Выделение аномальных последовательностей системных вызовов и LSM-событий для раннего обнаружения бесфайловых техник [7, 11].

3. Графотемпоральная модель зоны риска.

Шаги определения зон риска показаны на рис. 1, *а–в*.

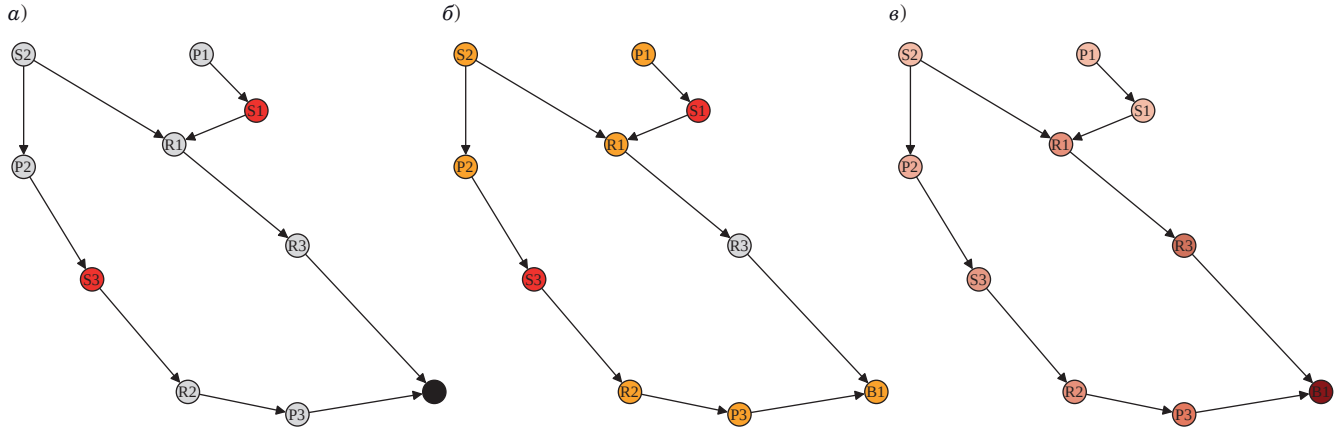
Серые узлы — обычные элементы графа (процессы, ресурсы, syscalls), которые пока считаются безопасными.

Красные узлы — якоря риска, найденные по логическим правилам над событиями ядра. В примере это два системных вызова S1 и S3, которые указывают на возможные признаки бесфайловой атаки (например, memfd_create и fexecve).

Черный узел — граница риска, т. е. точка, за которой находится критичный объект. В нашем случае это может быть сокет Docker (/var/run/docker.sock) или host namespace.

На этом этапе есть понимание только того, где начинается зона опасности (якоря) и где граница, до которой нельзя позволить достигнуть.

Оранжевые узлы — новые вершины, добавленные в процессе охраняемого замыкания (guarded closure). Узлы добавляются, если они находятся в «смысловой» близости к якорям: тот же контейнер, связанные ресурсы или syscalls,



■ ***Рис. 1.*** Графотемпоральная модель зоны риска: *а* — исходный граф с якорями; *б* — фронт замыкания; *в* — тепловая карта риска

■ ***Fig. 1.*** Graph-temporal risk zone model: *а* — initial graph with anchors; *б* — closure front; *в* — risk heat map

ведущие к границе. Полученный подграф образует структурный контур зоны риска, т. е. минимальную окрестность, достаточную для представления всех потенциальных причинно-временных цепочек атаки, которые в дальнейшем могут привести к достижению опасных границ.

Узлы на рис. 1, *в* окрашены по интенсивности красного цвета в зависимости от значения потенциала риска $u(v)$. Чем краснее, тем выше вероятность, что этот узел «дойдет» по допустимым ребрам до границы (в примере — B1). Чем светлее, тем ниже риск — узел либо далеко, либо связан через безопасные пути. Потенциал вычислен через поглощающие случайные блуждания (absorbing random walk): поглощающие состояния — это границы риска, а веса ребер зависят от их «опасности» (syscalls вроде setns, mmap, bpf дают высокий вес). Далее переходим от «геометрии» зоны (замыкание) к картине распределения риска. Это позволяет отсечь шум и выделить только те узлы, которые реально могут привести к атаке. При выборе порога θ получаем финальный контур зоны риска.

Данный способ отличается от стандартных методов, использующих полный граф или сигнатурные правила. В научной литературе встречаются подходы на основе графов зависимостей (dependency graphs), происхождения данных (provenance graphs) [17] и GNN-моделей [18, 19]. Однако подход, в котором зона риска сначала формируется с помощью охраняемого темпорального замыкания (guarded closure), а затем ее граница уточняется с использованием поглощающего случайного блуждания с ограничениями по меткам (label-constrained absorbing random walk), в рассмотренной нами литературе не выявлен. Метод обладает устойчивостью к шуму, так как замыкание гарантирует полноту включения релевантных узлов, а расчет потенциала отсекает малозначимые ветви.

Математическая модель зоны риска

Поток событий. Пусть $E_{evt} = \{e_i\}_{i=1}^{N}$ — упорядоченный по времени поток событий. Каждое ядровое событие задаем кортежем

$$e_i = (p_i, s_i, a_i, r_i, c_i, t_i),\ i = 1, \ldots, N, \qquad (1)$$

где p_i — процесс (PID, имя, контейнерный PID); $s_i \in S$ — системный вызов или LSM-событие (например, execve, memfd_create, setns); $a_i \in A$ — релевантные аргументы вызова (дескриптор, флаги, маска capabilities); $r_i \in R$ — ресурс (файл, сокет, объект namespace и т. п.); $c_i \in C$ — контейнерный контекст (cgroup, pod, image, набор событий); $t_i \in R^+$ — время события.

Граф исполнения. Из потока событий строим темпоральный атрибутированный ориентированный мультиграф

$$G = (V, E, \lambda, \tau), \qquad (2)$$

где V — множество вершин: процессы, ресурсы, пространства имен, состояния привилегий; $E \subseteq V \times V$ — мультимножество ориентированных ребер (причинно-временные связи между узлами); λ: $E \to \Lambda$ — метка ребра (тип вызова, объем байтов, код возврата, изменения capabilities); τ: $E \to R_{>0}$ — функция временных меток (время события).

Границы. Критические вершины (границы риска) задаем множеством

$$B \subseteq V, \qquad (3)$$

это вершины, достижение которых нарушителем считается критическим (например, /var/run/docker.sock, containerd.sock, host-namespace, состояния с CAP_SYS_ADMIN).

Якоря риска — событие $e \in E_{evt}$, удовлетворяющее булевой формуле $\Phi(e)$ над предикатами. В предыдущем разделе были перечислены основные способы атаки, отражающие ключевые приемы бесфайловых атак на Linux-контейнеры. Для формализации математической модели эти способы объединены в шесть формальных предикатов, каждый из которых может включать несколько исходных способов атаки, предложенных в табл. 1.

В случае появления новых способов бесфайловых атак модернизация формальных предикатов в математической модели не представляет сложности, так как любой новый способ возможно отнести к одному из уже определенных предикатов, что обеспечивает гибкость модели и упрощает ее расширение для поддержки новых угроз.

Множество якорей

$$A = \{e \in E_{evt} | \Phi(e) = 1\}. \qquad (4)$$

Охраняемое темпоральное замыкание. Чтобы из множества «точек-якорей» получить структурный контур зоны риска, мы распространяем влияние по графу исполнения, ограничивая допустимые переходы как по смыслу, так и по времени. Процедура состоит из трех логически связанных шагов.

1. *Охрана ребер*. Зададим предикат

$$\Gamma: E \to \{0, 1\}, \qquad (5)$$

который разрешает распространяться только по допустимым ребрам (например, в том же контейнере; или явные переходы границы с фик-

■ ***Таблица 1.*** Способы бесфайловых атак и формальные предикаты
■ ***Table 1.*** Fileless attack techniques and formal predicates

Исходный способ атаки	Формальный предикат
Использование временных файловых систем для размещения и исполнения кода в памяти без записи на диск	P_{mem}
Применение системного вызова memfd_create() для создания анонимных файлов и исполнения исполняемых файлов	P_{mem}
Злоупотребление механизмом LD_PRELOAD для внедрения библиотек и перехвата вызовов функций процесса	P_{mem}
Использование ptrace для отладки и инъекции кода в другие процессы, обхода защитных механизмов	P_{inj}
Активное применение LOLBins (GTFObins) — выполнение команд, подключений, эскалации и побега из контейнера	P_{sock}
Эксплуатация доступа к сокетам контейнерного рантайма и namespace-настроек для латерального перемещения	P_{sock}, P_{ns}
Извлечение и использование секретов Linux capabilities для закрепления и дальнейшего распространения атаки	P_{cap}
Аномальные последовательности системных вызовов и LSM-событий	P_{bpf}

сацией capabilities). На основе предиката Γ для произвольного множества узлов $S \subseteq V$ вводится оператор охраняемого темпорального замыкания.

2. *Оператор замыкания*. Для множества узлов $S \subseteq V$ определим

$$C^{k,\Delta}_{\Gamma,L}(S) = S \bigcup \{v \in V \mid \exists u \in S : u \overset{<k,\Delta}{\underset{\Gamma,L}{\Rightarrow}} v\}. \qquad (6)$$

Здесь $k \in N$ — радиус по ребрам (максимальное число переходов в путях), чем больше k, тем выше полнота, но возврат затраты и шум; $\Delta \in R$ — максимально допустимое «временное растяжение» пути (секунды — та же единица, что у t_i).

3. *Скелет зоны риска* — наименьшая фиксточка (минимальное неподвижное множество) этого оператора, порожденная узлами якорей. Оператор $C_r^{k,\Delta}$ монотонен, а значит, существует минимальное неподвижное множество

$$Z^* = \mu SC_r^{k,\Delta}(S \bigcup nodes(A)), \qquad (7)$$

где $nodes(A) = \{src(e), dst(e) | e \in A\}$; $A \subset E_{evt}$ — якоря-события. После выделения якорей риска расширяем зону вокруг якорей так далеко, как это необходимо и достаточно, чтобы покрыть все потенциальные причинно-временные цепочки атаки в пределах k переходов и окна Δ, но не дальше.

Приведем ключевые свойства оператора: (i) C монотонен $\Rightarrow$ μ-точка существует; (ii) оператор вычисляется итеративно локально вокруг A, и время оценивается как $O(k|E_{\text{лок}}|)$, где $E_{\text{лок}} \to \subseteq E$ — подмножество ребер, достижимых из $nodes(A)$ при $\Gamma(e) = 1$ и в пределах ограничений k, Δ; (iii) параметры k, Δ управляют «шириной» зоны и бюджетом телеметрии.

Поле риска на подграфе (выделение финальной зоны). Скелет Z^* может содержать «хвосты», которые структурно близки, но практически малоопасны. Поэтому вводим полевую оценку риска на индуцированном подграфе. Отделяем исходный мультиграф и индуцированный:

$$H = G[Z^*] = (V_H, E_H, \lambda|E_H, \tau|E_H), \qquad (8)$$

где $V_H = Z^*$ и $E_H = \{e \in E | src(e), dst(e) \in V_H\}$.

Далее формируем поглощающие случайные блуждания с ограничениями по меткам:

— веса ребер w: $E_H \to R_{>0}$: опасные метки (например, setns, mmap, bpf) получают больший вес;

— поглощение на границах: узлы $B \cap V_H$ делаем поглощающими (если блуждание дошло до границы, оно «исчезает» там);

— ограничение допустимых цепочек: фиксируем язык L, это регулярные паттерны TTP (Tactics, Techniques and Procedures) в терминологии MITRE ATT & CK. Запрещаем все переходы, которые не соответствуют L. Технически это можно реализовать как декартово произведение графа H с автоматом A_L.

В результате, решая стандартную задачу поглощения, получаем вектор вероятностей

$$\overline{\mathbf{u}} : V^H \to 0,1, \overline{\mathbf{u}}(v) = \Pr\{\text{дойти из } v \text{ до } B \bigcap V_H\}. \qquad (9)$$

Это и есть «потенциал риска»: чем ближе узел к опасным границам по допустимым ребрам, тем больше $\bar{\mathbf{u}}(v)$.

Граничные условия для потенциала риска:

1) множество поглощающих вершин: $B \cap V_H$ — узлы, достижение которых критично (границы риска). Для них фиксируем $\bar{\mathbf{u}}(v) = 1$;

2) множество безопасных опорных вершин (специальные вершины в графе исполнения, которые принимаются как «абсолютно безопасные» начальные точки для расчета потенциала риска): $S_{safe} \subseteq V_H$ — вершины, гарантированно не вовлеченные в атаку (например, долгоживущие системные процессы, инфраструктурные демоны). Для них фиксируем $\bar{\mathbf{u}}(v) = 0$;

3) для остальных вершин $\mathbf{u}$ определяется как решение задачи гармоничности

$$\bar{\mathbf{u}}(v) = \frac{\sum_{(v,x)\in E_H} w(v, x) * \bar{\mathbf{u}}(x)}{\sum_{(v,x)\in E_H} w(v, x)},$$
$$v \in V_H (B \cup S_{safe}). \qquad (10)$$

Финальная зона риска — уровневое множество потенциала

$$Z_\theta = \{v \in V^H \mid \bar{\mathbf{u}}(v) \geq \theta\}, \qquad (11)$$

где $\theta \in (0, 1)$ — заданный порог (калибруется по ROC/PR на валидации).

Эквивалентно можно решить дискретную задачу Дирихле для взвешенного лапласиана на H: задать $\mathbf{u} = 1$ на B, $\mathbf{u} = 0$ на безопасных «сидерах» (например, долгоживущие системные процессы) и получить гармоническое поле $\mathbf{u}$; затем порогом θ выделить Z_θ.

Сигнал тревоги подается после вычисления H, u, Z_θ, если выполнены следующие условия:

1) существуют $a \in A_v$ и $b \in B \cap V_H$ такие, что внутри H

$$a \overset{\Gamma,\ L}{\underset{<k,\Delta}{\Rightarrow\Rightarrow}} b; \qquad (12)$$

2) $\max_{v \in A_v} u(v \geq \theta)$.

Испытательный стенд

Практическая часть работы проводилась на операционной системе Ubuntu (24.04.3 LTS), с развернутым Kubernetes (v1.32, containerd v2.0) кластером в конфигурации один мастер и два воркер-сервера. В качестве сетевого плагина использован Cilium. Все LSM-события и системные вызовы собраны с помощью Tetragon, который в свою очередь использует eBPF и развернут в виде DaemonSet (privileged, hostPID, hostNetwork). В Tetragon заданы минимальные правила, которые не блокируют, а только логируют указанные вызовы с аргументами и контекстом. Для контроля качества параллельно запускали Falco с включенными правилами “fileless via memfd_create” и “open docker.sock”. Falco похож на Tetragon, но регистрирует только системные вызовы (syscalls). В качестве целевого образа контейнера был взят nginx:1.25-alpine. Этот образ широко используется в промышленной эксплуатации и содержит типичный набор утилит BusyBox. Построение графа и зон риска реализовано на Python 3.11.

Паттерны S1–S6 представляют собой заведомо подозрительное поведение и являются якорями, B1–B2 — доброкачественные эпизоды (команды администрирования, статика NGINX, сетевой шум без доступа к границам). В табл. 2 сведены все сценарии генерации телеметрии.

Для каждого сценария проведено по 60 повторов с разными задержками, вставками безвредных вызовов и перестановками (эвазивность) и по 100 повторов фоновой активности. Итого 580 эпизодов, из них 240 с достижением границы риска и 340 без достижения границ. Для каждого повтора S-сценария мы случайно выбирали:

1) задержки между вызовами из диапазона 0,1–2,0 с;

2) вставки до $m = 5$ нейтральных системных вызовов из словаря {stat, read, getpid и др.};

3) перестановки соседних блоков длиной 1–3 вызова. Параметр сложности эвазии обозначаем k (число модификаций).

Эпизод получает метку 1, если в полном графе событий эпизода действительно зафиксиро-

■ ***Таблица 2.*** Сценарии генерации телеметрии
■ ***Table 2.*** Telemetry generation scenarios

ID	TTP-паттерн	Правила генерации (упрощенно)
S1	memfd → exec	memfd_create → fexecve в ≤5s; ресурс memfd://…
S2	X∧W	mmap/mprotect с prot = X
S3	Ptrace	Одиночный ptrace_attach к целевому pid в том же pod
S4	/dev/shm	write в /dev/shm/foo → execve / tmp/x (обычный true)
S5	setns	setns PID/MOUNT → mount read-only
S6	docker.sock	open /var/run/docker.sock (без сокетного обмена)
B1	Benign web	open/read/write статики nginx
B2	Benign batch	fork/exec воркера, сетевые подключения к базе данных

вано достижение границы риска: открыт docker.sock/containerd.sock, выполнен setns в host-namespace, получена повышающая capability и другие подобные ситуации. Во всех прочих случаях метка 0. Разметка производится автоматически по событиям ядра и проверяется вручную на случай неоднозначностей. Разбиение на тренировочную и валидационную выборку случайное.

Построение графа и зоны риска состоит из четырех этапов.

1. Формирование графа исполнения. Для каждого эпизода из сырых eBPF-событий сформировали гетерогенный граф исполнения. Узлы представляли процессы, файлы/сокеты, пространства имен и состояния привилегий; ребра кодировали факты взаимодействия. Параллельно строился граф системных вызовов, фиксирующий причинные последовательности по PID/TID (Process/Thread ID) и времени. Такой двойной взгляд (HG+SCG) сохраняет и структуру ресурсов, и темпоральную причинность.

2. Локализация зоны вокруг якорей. Из множества якорей индуцировали локальную зону риска, расширяя граф от якорей на ограниченную глубину и в заданном временном окне. Фильтрация по предикату релевантности исключала нерелевантные переходы (другие контейнеры, несвязанную активность). Результат — индуцированный подграф H, существенно меньший по размеру, но сохранивший достижимые пути к границам риска.

3. Ранжирование по потенциалу риска. На подграфе H вычислялся потенциал риска: границы риска (например, docker/containerd-сокеты, host-namespace, получение критичных capabilities) рассматривались как поглощающие вершины, ребра взвешивались по типу действия и контексту. Полученное поле значений интерпретировалось как степень достижимости границ из соответствующих вершин. Этот расчет выполнялся пакетно для каждого эпизода и не изменял исходную телеметрию.

4. Сводка в скор и оценка метрик. Для эпизода агрегировали потенциал в один числовой скор (максимум или высокий перцентиль по вершинам подграфа). Набор пар (score, label) по тестовой выборке использовали для построения PR/ROC-кривых и вычисления AUPRC/AUROC (площадь под PR-кривой / площадь под ROC-кривой), а также для оценки задержки детекции и устойчивости к эвазивности и потере событий.

Предложенный в статье метод рассматривался в сравнении с тремя базовыми методами:

1) статическими правилами (Falco/Tetragon): сигнатуры/правила преобразовывались в оценки через нормированную «важность» срабатывающих правил (максимум по эпизоду);

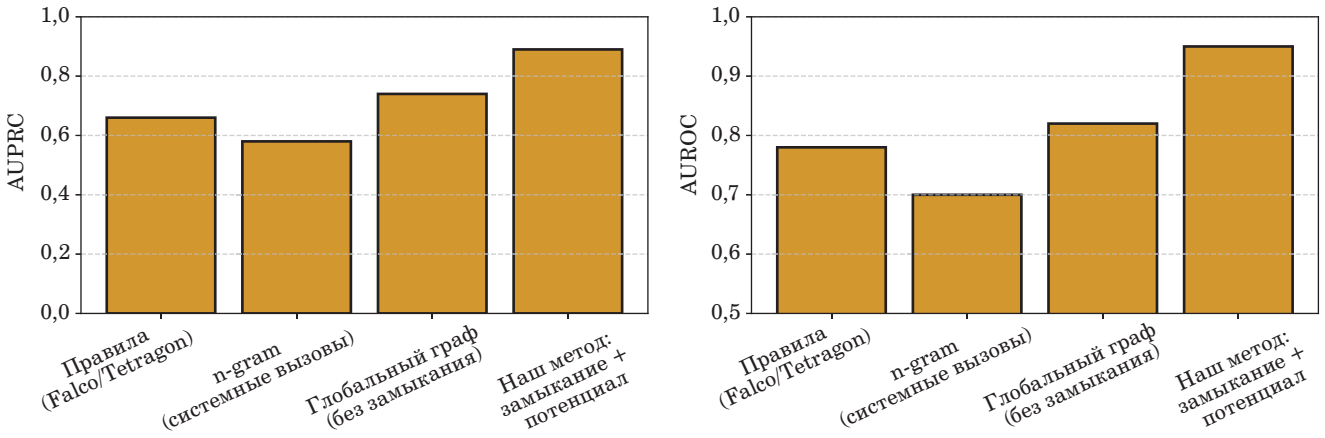
2) n-gram (syscalls): оценка «аномальности» последовательности системных вызовов. Рассматривали последовательность вызовов как язык и оценивали, насколько вероятен текст. Оценка — это мера маловероятности эпизода с последующей нормализацией [0, 1];

3) Global HG: тот же потенциал u, но на полном графе G без локализации замыканием.

Результаты

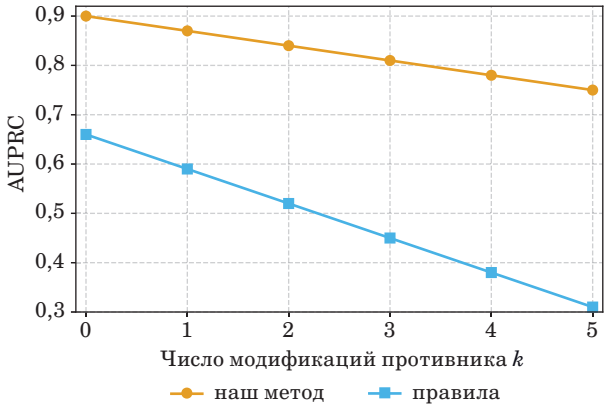
На тестовой выборке предложенный риск-центричный метод показал AUPRC = 0,87 и AUROC = 0,94, что превосходит все базовые подходы (рис. 2).

Атакующие могут маскироваться и выполнять легитимные вызовы между якорями, менять порядок нейтральных действий и добавлять задержки. Эти действия моделируются как



■ ***Рис. 2.*** Сравнение метрик AUPRC и AUROC по методам

■ ***Fig. 2.*** Comparison of metrics AUPRC and AUROC by methods



■ ***Рис. 3.*** Сравнение робастности к эвазивности
■ ***Fig. 3.*** Comparison of robustness against evasion

k-модификаций в эпизоде. На рис. 3 показано снижение качества детекции при усложнении (маскировке) действий противника.

При росте числа модификаций противника k качество нашего метода падает значительно медленнее: AUPRC снижается с 0,90 до 0,75 (≈ 17 %), тогда как для правил — с 0,66 до 0,31 (≈ 53 %). Это подтверждает робастность локализованного анализа «зоны риска» к эвазивным изменениям и сохраняет практическую точность даже при $k = 5$. На практике это означает более стабильные оповещения без постоянной перенастройки правил при попытках обхода детектирования.

DaemonSet Tetragon на рабочем узле потреблял 1,5 % CPU (медиана; p95 — 3,5 %), память 180 мегабайт при средней интенсивности около 1200 событий в секунду. Построение локального подграфа H из сырых eBPF-событий занимало в среднем 12 мс на эпизод. Суммарная задержка формирования скора эпизода (парсинг $\rightarrow H \rightarrow u \rightarrow$ агрегация) составляла 36 мс.

Обсуждение

Исследование [20] по n-gram/STIDE/BoSC для Docker/LXC фиксирует высокие показатели точности (до 97 %), однако эти подходы чувствительны к эвазии через вставки нейтральных вызовов и не моделируют достижимость именно контейнерных границ риска. Наши эксперименты прямо оценивают падение качества при эвазии, что делает сравнение на этой оси прозрачным. В работе [11] рассматривается онлайн-детектор атак в контейнерах на частотах syscalls с самосупервизией. Этот подход показывает уменьшение ложных тревог на 33–93 % по сравнению с наборами классических ML-схем при сопоставимой полноте. В отличие от данных частотных/оконных признаков, наш метод строит гетерогенный HG/SCG-граф из eBPF-телеметрии и агрегирует «достижимость» реально опасных границ, что повышает переносимость между сервисами при сохранении интерпретации причинных путей.

Сравнивая предложенный риск-центричный метод с известными подходами к обнаружению бесфайловых атак в контейнерах, можно выделить следующие особенности.

1. Статические и сигнатурные методы (IDS, SIEM, антивирусы) малоэффективны против бесфайловых угроз и дают большое количество ложных срабатываний при динамической нагрузке, в то время как предложенный метод анализирует локальные подграфы зон риска и учитывает контекст контейнера, что повышает точность детекции.

2. Анализ оперативной памяти (memory forensics) требует сложных дампов и ручной обработки, тогда как предложенный метод работает онлайн на основе потоковых eBPF-событий и не блокирует контейнеры.

3. Модели на основе системных вызовов (Bag-of-Syscalls) чувствительны к шуму и изменениям нагрузки; риск-центричный подход снижает влияние шума за счет выделения локальных зон риска и учета временных параметров событий.

4. eBPF-сенсоры и kernel-based-мониторинг обеспечивают сбор событий ядра, но могут повышать нагрузку при высокой интенсивности. Предложенный в статье метод также использует eBPF, но за счет локализации зон риска и ограниченного построения графа сохраняет производительность и масштабируемость.

5. Графовые модели поведения (ориентированные графы системных вызовов, GNN, логические графы атак) являются наиболее близкими к предложенному подходу. В отличие от Container Escape Detection, Phoenix и CORAL [14] разработанный риск-центричный метод обнаружения бесфайловых угроз в контейнерных средах:

— не требует полного построения графа всех событий, снижая чувствительность к шуму и нагрузку на ресурсы;

— ориентирован на детекцию бесфайловых (in-memory) атак, а не только на заранее известные последовательности или escape-сценарии;

— учитывает динамический контекст контейнера и временные зависимости, что повышает устойчивость при неполном мониторинге.

Таким образом, предложенный метод сочетает преимущества графового подхода и динамического анализа, обеспечивая интерпретируемость, устойчивость к шуму и масштабируемость, что делает его более применимым в современных контейнерных инфраструктурах по сравнению с существующими решениями.

Заключение

В работе предложен риск-центричный метод обнаружения бесфайловых угроз в контейнерных средах, который смещает фокус от анализа полного графа происхождения событий к выделению локальных подграфов «зон риска», индуцируемых eBPF-телеметрией и заданными якорями. В основе метода лежит графотемпоральная модель исполнения, где события ядра и системные вызовы представляются в виде гетерогенного графа. Для локализации потенциальных областей атаки введены формальные предикаты охраны ребер, ограничивающие возможные траектории распространения, и оператор замыкания, формирующий скелет зоны риска вокруг якорей.

Для оценки подготовлено 580 эпизодов поведения контейнера nginx:1.25-alpine: 240 с достижением границы риска (переходы к docker.sock/containerd.sock, host-namespace, повышающие capabilities и др.) и 340 без достижения; для каждого сценария S1–S6 выполнено по 60 повторов с эвазивными модификациями плюс 100 повторов фоновой активности.

По сравнению с подходами, основанными на сигнатурных правилах Falco/Tetragon, n-gram по системным вызовам и глобальном графе без локализации, предлагаемый метод показывает преимущество по AUPRC = 0,87 и AUROC = 0,94, а также более медленное падение качества при эвазивных k-модификациях.

Практическая ценность — в интерпретируемости (контур зоны риска, $u(v)$) и снижении ложных срабатываний, что важно для раннего реагирования.

Научная новизна работы заключается в применении риск-центричного подхода к анализу поведения контейнеров с учетом временных зависимостей событий, что позволяет локализовать зоны риска и повышает устойчивость обнаружения бесфайловых атак.

Таким образом, предложенная строгая математическая модель и практическая реализация на реальном кластере Kubernetes формируют интерпретируемую и устойчивую основу в системы ИБ, подтверждая применимость риск-центричного метода для обнаружения актуальных бесфайловых угроз в условиях современных контейнерных инфраструктур.

Дальнейшие исследования включают открытые бенчмарки и трассы промышленной эксплуатации с более широким покрытием TTP; адаптивную калибровку порога θ и весов ребер под стоимость ошибок (cost-sensitive ROC/PR); комбинирование локальной зоны риска с легкими графовыми нейросетевыми приор-оценками на этапе ранжирования.

Литература

1. **Liu S., Peng G., Zeng H., Fu J.** A survey on the evolution of fileless attacks and detection techniques. *Computers & Security*, 2024, vol. 137, Art. 103653. doi:10.1016/j.cose.2023.103653/issn0167-4048
2. **Doherty A., Zigdon Y., Ron A.** *Aqua Nautilus Research Finds 1,400% Surge in Memory-Based Attacks as Hackers Evade Traditional Cloud Security Defenses. Aqua Security Research Report*. 2023. https://www.aquasec.com/news/aqua-nautilus-research-finds-1400-surge-in-memory-based-attacks/ (дата обращения: 24.07.2025).
3. **Sudhakar, Kumar S.** An emerging threat Fileless malware: A survey and research challenges. *Cybersecurity*, 2020, vol. 3, Art. 1. doi:10.1186/s42400-019-0043-x
4. **Kara I.** Fileless malware threats: Recent advances, analysis approach through memory forensics and research challenges. *Expert Systems with Applications*, 2023, vol. 214, Art. 119133. doi:10.1016/j.eswa.2022.119133
5. **Wu M.-H., Hsu F.-H., Huang J.-H., Wang K., Hwang Y.-L., Wang H.-J., Chen J.-X., Hsiao T.-C., Yang H.-T.** Enhancing linux system security: A kernel-based approach to fileless malware detection and mitigation. *Electronics*, 2024, vol. 13, Art. 3569. doi:10.3390/electronics13173569
6. **Levy Rocha Savio, Lopes de Mendonca Fabio Lucio, Staciarini Puttini Ricardo, Rabelo Nunes Rafael, Amvame Nze Georges Daniel Less**. DCIDS — Distributed Container IDS. *MDPI AG*, 2023, vol. 13, iss. 16, Art. 9301. doi:10.3390/app13169301
7. **Amr S. Abed, T. Charles Clancy, David S. Levy.** Applying bag of system calls for anomalous behavior detection of applications in Linux containers. *IEEE Globecom Workshops (GC Wkshps)*, 2015, doi:10.1109/GLOCOMW.2015.7414047
8. **Ковалев М. Г.** Трассировка сетевых пакетов в ядре Linux с использованием eBPF. *Труды Института системного программирования РАН*, 2020, т. 32, вып. 3, с. 71–77. doi:10.15514/ISPRAS-2020-32(3)-6
9. **Котенко И. В., Мельник М. В.** Обнаружение аномалий в контейнерных системах: применение частотного анализа и гибридной нейронной сети. *Программные продукты и системы*, 2025, т. 38, № 3, с. 426–437. doi:10.15827/0236-235X.151.426-437
10. **Pope J., Liang J., Kumar V., Raimondo F., Sun X., McConville R., Pasquier T., Piechocki R., Oikonomou G., Luo B., Howarth D., Mavromatis I., Sanchez Mompo A., Carnelli P., Spyridopoulos T., Khan A.** Resource-interaction graph: Efficient graph representation for anomaly detection. *arXiv preprint*, 2022. doi:10.48550/arXiv.2212.08525

11. **Tunde-Onadele O., Lin Y., Gu X., He J., Latapie H.** Self-supervised machine learning framework for online container security attack detection. *ACM Transactions on Autonomous and Adaptive Systems*, 2024, vol. 19, iss. 3. doi:10.1145/3665795
12. **Shokouhinejad H., Razavi-Far R., Mohammadian H., Rabbani M., Ansong S., Higgins G., Ghorbani A. A.** Recent advances in malware detection: Graph learning and explainability. *arXiv preprint IEEE*, 2025. arXiv:2502.10556. doi:10.48550/arXiv.2502.10556
13. **Chen K., Zhao Y., Guo J., Gu Z., Han L., Tang K. A**. Container escape detection method based on a dependency graph. *Electronics*, 2024, vol. 13, no. 23, Art. 4773. doi:10.3390/electronics13234773
14. **Tayouri D., Sgan Cohen O., Maimon I., Mimran D., Elovici Y., Shabtai A.** CORAL: Container Online Risk Assessment with Logical attack graphs. *Computers & Security*, 2025, vol. 150, iss. C, Art. 104296. doi:10.1016/j.cose.2024.104296
15. **Boros T., Cotaie A., Stan A., Vikramjeet K., Malik V., Davidson J.** Machine learning and feature engineering for detecting living off the land attacks. *Proceedings of the 7th International Conference on Internet of Things, Big Data and Security (IoTBDS 2022)*, 2022, pp. 133–140. doi:10.5220/0011004500003194
16. **Kermabon-Bobinnec H., Jarraya Y., Wang L., Majumdar S., Pourzandi M.** Phoenix: Surviving unpatched vulnerabilities via accurate and efficient filtering of syscall sequences. *Network and Distributed System Security Symposium (NDSS) 2024*, San Diego, USA, February 26–March 1, 2024. doi:10.14722/ndss.2024.24582
17. **Milajerdi S. M., Gjomemo R., Eshete B., Sekar R., Venkatakrishnan V. N.** HOLMES: Real-time APT detection through correlation of suspicious information flows. *IEEE Symposium on Security and Privacy*, San Francisco, CA, USA, 2019, pp. 1137–1152. doi:10.1109/SP.2019.00026
18. **Ziqi L., Chaochao C., Xinxing Y., Jun Z., Xiaolong L., Le S.** Heterogeneous graph neural networks for malicious account detection. *arXiv preprint*, 2020. arXiv:2002.12307. doi:10.48550/arXiv.2002.12307
19. **Haoyu Wu, Dongyu L., Luxin Z., Xiaoshan Z., Jiajun Z., Shanqing Yu, Qi X.** A graph neural network framework for dynamic malware detection using api calls and lightweight containers. *2025 5th International Conference on Intelligent Communications and Computing (ICICC)*, Nanjing, China, 2025, 15–17 August. doi:10.1109/ICICC66840.2025.11199642
20. **Castanhel G. R., Heinrich T., Ceschin F., Maziero C.** Taking a peek: An evaluation of anomaly detection using system calls for containers. *Conference: 26th IEEE Symposium on Computers and Communications (ISCC)*, Athens, Greece, 2021, 5–8 September, IEEE, 2021. doi:10.1109/ISCC53001.2021.9631251

UDC 004.056:004.75
doi:10.31799/1684-8853-2026-1-48-60
EDN: AVYJSY

Using graphs to detect fileless attacks in containerized infrastructure

E. O. Zdornikov[a], Programmer Engineer, orcid.org/0009-0009-0154-5153
E. V. Egorov[b], Junior Researcher, orcid.org/0009-0006-4321-0069
I. Y. Popov[a], PhD, Tech., Associate Professor, orcid.org/0000-0002-6407-7934, ilyapopov27@gmail.com
[a]ITMO University, 49, Kronverksky Pr., 197101, Saint-Petersburg, Russian Federation
[b]A. F. Mozhaiskii Military Space Academy, 13, Zhdanovskaia Emb., 197198, Saint-Petersburg, Russian Federation

Introduction: Fileless attacks exploit memory to execute malicious code without storing artifacts on disk. This significantly complicates their detection and makes traditional protection methods ineffective, especially in dynamic containerized environments where processes frequently start and terminate automatically, generating additional noise. Container systems such as Docker and Kubernetes have a smaller volume of monitored data compared to virtual machines, which simplifies event analysis and activity graphing. **Purpose:** To develop a method for detecting fileless attacks in containerized infrastructures that remains robust to noise and evasive techniques under incomplete monitoring and provides early alerting before critical resources are reached. **Results:** We propose a risk-oriented method based on a heterogeneous graph and a system call graph. Risk zones are identified around events logged using eBPF, after which a mathematical model of risk zones with secure closure and risk potential calculation based on random walk absorption is formed. The model additionally takes into account the container context and timing parameters, which reduces the number of false positives. The method enables the localization of risk zones and operates reliably even with the loss of some events. Experiments have been conducted on a Kubernetes cluster (v1.32) running Ubuntu 24.04 using Tetragon (eBPF) and Falco for quality control. We have collected 580 container behavior episodes, including attack and background scenarios, and have then used them to generate heterogeneous execution and system call graphs. The proposed method outperforms static rules, n-gram system call models, and global graph methods in terms of AUROC and AUPRC metrics, demonstrating increased resilience to invasive techniques. **Practical relevance:** The developed method is compatible with existing sensors and container security policies. It provides system administrators with interpretable risk zones and quantitative indicators of attack probability, while also supporting integration into existing incident response systems. **Discussion:** The presented mathematical model and practical implementation confirm the applicability of risk-centric analysis for detecting contemporary fileless threats in modern containerized environments; the approach is scalable, parameterizable, and does not require application modification, which makes it suitable for deployment in industrial clusters.

Keywords — fileless attacks, container security, eBPF, graph-temporal analysis, runtime detection, system call graphs, Kubernetes.

For citation: Zdornikov E. O., Egorov E. V., Popov I. Y. Using graphs to detect fileless attacks in containerized infrastructure. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2026, no. 1, pp. 48–60 (In Russian). doi:10.31799/1684-8853-2026-1-48-60, EDN: AVYJSY

References

1. Liu S., Peng G., Zeng H., Fu J. A survey on the evolution of fileless attacks and detection techniques. *Computers & Security*, 2024, vol. 137, Art. 103653. doi:10.1016/j.cose.2023.103653/issn0167-4048
2. Doherty A., Zigdon Y., Ron A. *Aqua Nautilus Research Finds 1,400% Surge in Memory-Based Attacks as Hackers Evade Traditional Cloud Security Defenses. Aqua Security Research Report*. 2023. https://www.aquasec.com/news/aqua-nautilus-research-finds-1400-surge-in-memory-based-attacks/ (дата обращения: 24.07.2025).
3. Sudhakar, Kumar S. An emerging threat Fileless malware: A survey and research challenges. *Cybersecurity*, 2020, vol. 3, Art. 1. doi:10.1186/s42400-019-0043-x
4. Kara I. Fileless malware threats: Recent advances, analysis approach through memory forensics and research challenges. *Expert Systems with Applications*, 2023, vol. 214, Art. 119133. doi:10.1016/j.eswa.2022.119133
5. Wu M.-H., Hsu F.-H., Huang J.-H., Wang K., Hwang Y.-L., Wang H.-J., Chen J.-X., Hsiao T.-C., Yang H.-T. Enhancing linux system security: A kernel-based approach to fileless malware detection and mitigation. *Electronics*, 2024, vol. 13, Art. 3569. doi:10.3390/electronics13173569
6. Levy Rocha Savio, Lopes de Mendonca Fabio Lucio, Staciarini Puttini Ricardo, Rabelo Nunes Rafael, Amvame Nze Georges Daniel Less. DCIDS — Distributed Container IDS. *MDPI AG*, 2023, vol. 13, iss. 16, Art. 9301. doi:10.3390/app13169301
7. Amr S. Abed, T. Charles Clancy, David S. Levy. Applying bag of system calls for anomalous behavior detection of applications in Linux containers. *IEEE Globecom Workshops (GC Wkshps)*, 2015, doi:10.1109/GLOCOMW.2015.7414047
8. Kovalev M. G. Network packet tracing in the Linux kernel using eBPF. *Proceedings of the Institute for System Programming of the Russian Academy of Sciences*, 2020, vol. 32, iss. 3, pp. 71–77 (In Russian). doi:10.15514/ISPRAS-2020-32(3)-6
9. Kotenko I. V., Melnik M. V. Anomaly detection in container systems: application of frequency analysis and hybrid neural network. *Software & Systems*, 2025, vol. 38, no. 3, pp. 426–437 (In Russian). doi:10.15827/0236-235X.151.426-437
10. Pope J., Liang J., Kumar V., Raimondo F., Sun X., McConville R., Pasquier T., Piechocki R., Oikonomou G., Luo B., Howarth D., Mavromatis I., Sanchez Mompo A., Carnelli P., Spyridopoulos T., Khan A. Resource-interaction graph: Efficient graph representation for anomaly detection. *arXiv preprint*, 2022. doi:10.48550/arXiv.2212.08525
11. Tunde-Onadele O., Lin Y., Gu X., He J., Latapie H. Self-supervised machine learning framework for online container security attack detection. *ACM Transactions on Autonomous and Adaptive Systems*, 2024, vol. 19, iss. 3. doi:10.1145/3665795
12. Shokouhinejad H., Razavi-Far R., Mohammadian H., Rabbani M., Ansong S., Higgins G., Ghorbani A. A. Recent advances in malware detection: Graph learning and explainability. *arXiv preprint IEEE*, 2025. arXiv:2502.10556. doi:10.48550/arXiv.2502.10556
13. Chen K., Zhao Y., Guo J., Gu Z., Han L., Tang K. A. Container escape detection method based on a dependency graph. *Electronics*, 2024, vol. 13, no. 23, Art. 4773. doi:10.3390/electronics13234773
14. Tayouri D., Sgan Cohen O., Maimon I., Mimran D., Elovici Y., Shabtai A. CORAL: Container Online Risk Assessment with Logical attack graphs. *Computers & Security*, 2025, vol. 150, iss. C, Art. 104296. doi:10.1016/j.cose.2024.104296
15. Boros T., Cotaie A., Stan A., Vikramjeet K., Malik V., Davidson J. Machine learning and feature engineering for detecting living off the land attacks. *Proceedings of the 7th International Conference on Internet of Things, Big Data and Security (IoTBDS 2022)*, 2022, pp. 133–140. doi:10.5220/0011004500003194
16. Kermabon-Bobinnec H., Jarraya Y., Wang L., Majumdar S., Pourzandi M. Phoenix: Surviving unpatched vulnerabilities via accurate and efficient filtering of syscall sequences. *Network and Distributed System Security Symposium (NDSS) 2024*, San Diego, 2024. doi:10.14722/ndss.2024.24582
17. Milajerdi S. M., Gjomemo R., Eshete B., Sekar R., Venkatakrishnan V. N. HOLMES: Real-time APT detection through correlation of suspicious information flows. *IEEE Symposium on Security and Privacy*, San Francisco, CA, USA, 2019, pp. 1137–1152. doi:10.1109/SP.2019.00026
18. Ziqi L., Chaochao C., Xinxing Y., Jun Z., Xiaolong L., Le S. Heterogeneous graph neural networks for malicious account detection. *arXiv preprint*, 2020. arXiv:2002.12307. doi:10.48550/arXiv.2002.12307
19. Haoyu Wu, Dongyu L., Luxin Z., Xiaoshan Z., Jiajun Z., Shanqing Yu, Qi X. A graph neural network framework for dynamic malware detection using api calls and lightweight containers. *2025 5th International Conference on Intelligent Communications and Computing (ICICC)*, Nanjing, China, 2025. doi:10.1109/ICICC66840.2025.11199642
20. Castanhel G. R., Heinrich T., Ceschin F., Maziero C. Taking a peek: An evaluation of anomaly detection using system calls for containers. *Conference: 26th IEEE Symposium on Computers and Communications (ISCC)*, Athens, Greece, 2021, IEEE, 2021. doi:10.1109/ISCC53001.2021.9631251

УДК 004.056.53
doi:10.31799/1684-8853-2026-1-61-76
EDN: OIUQDB

Научные статьи
Articles

Методика противодействия сетевой разведке объекта критической информационной инфраструктуры злоумышленником на основе IoC-анализа

А. А. Шевченко*[а], *канд. техн. наук, доцент, orcid.org/0000-0001-9113-1089
В. А. Липатников*[б], *доктор техн. наук, профессор, orcid.org/0000-0002-3736-4743, lipatnikovanl@mail.ru
В. А. Задбоев*[б], *младший научный сотрудник, orcid.org/0009-0003-9362-1307
П. И. Кузин*[в], *канд. техн. наук, доцент, orcid.org/0000-0003-0880-6204
[а]*Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича, Большевиков пр., 22-1, Санкт-Петербург, 193232, РФ*
[б]*Военная академия связи им. Маршала Советского Союза С. М. Буденного, Тихорецкий пр., 3, Санкт-Петербург, 194064, РФ*
[в]*Санкт-Петербургский государственный лесотехнический университет им. С. М. Кирова, Институтский пер., 5, Санкт-Петербург, 194021, РФ*

Введение: *развитие новых способов воздействия на объект критической информационной инфраструктуры со стороны злоумышленников побуждает к поиску актуальных методик противодействия.* ***Цель:*** *путем исследования способов реализации сетевой разведки разработать методику противодействия данному типу воздействия злоумышленника на основе IoC-анализа, которая позволит повысить оперативность обнаружения вероятных угроз объекта критической информационной инфраструктуры.* ***Результаты:*** *с использованием системного подхода проведен анализ механизмов реализации сетевой разведки (ICMP, TCP/UDP, ARP, DNS, SNMP) и инструментов (nmap, arp-scan, DNSenum, snmpwalk) для выявления активных устройств, открытых портов, операционных систем и служб. Результаты анализа стали основой при разработке модели указанного типа атаки, позволившей установить конкретные IoC, которые возможно зафиксировать в сети на каждом этапе реализации данного воздействия злоумышленника. Синтез полученных знаний о ключевых IoC, таких как аномальные значения TTL, всплески ICMP и SYN-пакетов, повышенный DNS-трафик и повторные попытки аутентификации и способов обеспечения защищенности сети, дал возможность разработать методику противодействия сетевой разведке объекта критической информационной инфраструктуры на основе анализа IoC и ряд инструментальных подходов для оперативного выявления аномалий сетевого трафика и попыток несанкционированного доступа с использованием Python-библиотеки Scapy. Реализация отдельных этапов предложенной методики в виде программного обеспечения способствовала проведению анализа ее результативности в ряде экспериментов.* ***Практическая значимость:*** *определяется возможностью использовать предложенную методику при разработке информационно-управляющих систем обеспечения информационной безопасности объектов критической информационной инфраструктуры.*

Ключевые слова *— информационная безопасность, объект критической информационной инфраструктуры, несанкционированный доступ, анализ трафика, IoC, модель, ранняя нейтрализация угроз.*

Для цитирования: Шевченко А. А., Липатников В. А., Задбоев В. А., Кузин П. И. Методика противодействия сетевой разведке объекта критической информационной инфраструктуры злоумышленником на основе IoC-анализа. *Информационно-управляющие системы*, 2026, № 1, с. 61–76. doi:10.31799/1684-8853-2026-1-61-76, EDN: OIUQDB
For citation: Shevchenko A. A., Lipatnikov V. A., Zadboev V. A., Kuzin P. I. Methodology based on the IoC analysis for countering attacker network reconnaissance on a critical information infrastructure facility. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2026, no. 1, pp. 61–76 (In Russian). doi:10.31799/1684-8853-2026-1-61-76, EDN: OIUQDB

Введение

Безопасная информационная инфраструктура является основой устойчивого функционирования государства и общества. Все более глубокая цифровизация этих систем делает их уязвимыми к компьютерным атакам (КА), последствия которых могут привести не только к экономическим потерям, но и к угрозе жизни людей.

Количество КА на российскую инфраструктуру с начала специальной военной операции увеличилось в разы, в том числе со стороны проправительственных хакерских групп Украины, которые атакуют российские предприятия оборонно-промышленного комплекса, также отмечается рост скорости исполнения атак. Поэтому разработка эффективных методик противодействия сетевой разведке объектов критической информационной инфраструктуры (КИИ) становится приоритетной задачей национальной и информационной безопасности (ИБ) [1–3]. В современных условиях реализация КА носит многоэтапный характер, причем в большинстве случаев невозможно определенно установить, в какой конкретный момент времени злоумыш-

ленник перешел к следующей стадии эскалации информационного конфликта. Одним из решений данной проблемы является внедрение в системы обеспечения ИБ объектов КИИ средств анализа индикаторов компрометации (Indication of Compromise, IoC). IoC — это признак, свидетельствующий о возможном инциденте ИБ в объекте КИИ, системе или устройстве, т. е. цифровой «след», по которому возможно выявить вредоносную активность на ранней стадии развития кризисной ситуации.

Ученые во всем мире активно разрабатывают способы к защите объектов КИИ с использованием указанного подхода. Так, в работах [4–6] предложили комплексные стратегии обеспечения ИБ, учитывающие внедрение передовых технологий, таких как искусственный интеллект, машинное обучение и блокчейн, в основные подпроцессы (аудит безопасности и реагирование на инциденты). Авторами в [7–9] описана концепция интеграции искусственного интеллекта и машинного обучения, в частности глубокого обучения, в систему обнаружения вторжений. В [10, 11] предлагается метод оценки уровня уязвимости сети от появления тех или иных IoC с использованием STIX-графа для структуризации информации об угрозах. Данные работы заложили основу для понимания природы угроз и путей обнаружения IoC.

Однако описанные подходы фокусируются на мониторинге и выявлении атак, уделяя меньше внимания практическим способам их безопасной нейтрализации. В связи с этим разработка и верификация комплексных методик противодействия сетевой разведке объекта КИИ, сочетающих технические, организационные и аналитические меры для обеспечения непрерывности и устойчивости критически важных процессов, является актуальным направлением исследования.

С учетом вышесказанного поставлена задача: исследовать способы реализации сетевой разведки, разработать методику противодействия указанной угрозы объекта КИИ путем раннего выявления IoC. Сетевая разведка выбрана ввиду того, что является одной из основных атак, с которой злоумышленник начинает воздействие на объект КИИ.

Моделирование сетевой разведки объекта КИИ и выявление IoC

Сетевая разведка — это один из видов КА на объекты КИИ, который заключается в исследовании сети в целях сбора информации о ее структуре, активных устройствах, открытых портах, запущенных службах и других характеристик [12, 13]. Это может быть как легитимное действие (например, при проведении аудита безопасности), так и злонамеренное (при подготовке к атаке).

Основные цели сетевой разведки:

1) обнаружение активных устройств — определение IP-адресов, MAC-адресов и других идентификаторов устройств в сети;

2) идентификация открытых портов — поиск портов, на которых работают сетевые службы (например, HTTP, FTP, SSH);

3) определение операционных систем и служб — установление типов операционных систем и версий программного обеспечения (ПО), работающих на устройствах;

4) построение карты сети — создание схемы сети, включая маршрутизаторы, коммутаторы, серверы и другие устройства;

5) выявление уязвимостей — поиск слабых мест в конфигурации сети или ПО.

Методы сетевой разведки включают различные подходы для исследования и анализа сетевой инфраструктуры. Одним из основных методов является перехват сетевого трафика, который позволяет наблюдать за обменом данными между узлами сети, выявлять используемые протоколы и типы передаваемой информации, скрывая свое присутствие в сети. В рамках этого метода используются специализированные анализаторы пакетов и средства мониторинга, обеспечивающие сбор, фильтрацию и интерпретацию трафика, например с помощью утилит *tshark* или *tcpdump*.

Следующим методом является ping-сканирование (ICMP-сканирование), которое используется для обнаружения активных устройств в сети. В рамках этого метода отправляются ICMP-запросы на целевые IP-адреса, что позволяет определить, какие устройства отвечают. Для выполнения таких задач часто применяются инструменты, такие как утилиты *ping* или *fping* [14].

Еще одним важным методом является сканирование портов, которое позволяет проверить состояние портов (открыт, закрыт или фильтруется). В рамках этого метода используются различные типы сканирования, например отправка TCP-пакетов с флагами SYN, ACK, FIN и др.

Также проводится UDP-сканирование, которое проверяет UDP-порты, часто используемые для таких служб, как DNS и DHCP [15]. Одним из популярных инструментов для сканирования портов является утилита *nmap*.

Для более глубокого анализа сети применяется сканирование операционных систем и служб. Этот метод позволяет определить тип операционной системы и версии запущенных служб на основе анализа ответов от устройств. Утилиты,

такие как *nmap -O* и *nmap -sV*, помогают в реализации этого подхода и поиске IoC.

В локальных сетях часто используется сканирование на основе ARP, которое позволяет обнаруживать устройства по их MAC-адресам. Для этого применяются такие инструменты, как *arp-scan* [16]. Также важным методом является сканирование на основе DNS, которое направлено на сбор информации о доменных именах, поддоменах и связанных IP-адресах. Инструменты, такие как *DNSenum* и *dig*, помогают в выполнении этой задачи.

Еще одним методом является сканирование на основе SNMP, которое использует протокол SNMP для получения информации о таких сетевых устройствах, как маршрутизаторы и коммутаторы [17, 18]. Для этого может использоваться инструмент *snmpwalk*.

Пример использования инструмента *nmap* представлен на рис. 1.

Для углубленного анализа сетевой разведки и выявления IoC, которые появляются в сети на каждом этапе реализации атаки, необходимо провести ее моделирование, например, с использованием модели, разработанной корпорацией Lockheed Martin для описания жизненного цикла КА Cyber Kill Chain [19]. В данной модели процесс реализации КА декомпозирован на семь этапов. В случае сетевого сканирования процесс реализации злоумышленником выглядит следующим образом:

1) разведка — пассивный сбор информации о сети, изучение активных хостов и сервисов;

2) вооружение — подготовка инструментов для сканирования сети;

3) доставка — активное сканирование сети;

4) заражение — анализ полученных данных и выявление уязвимостей;

5) инсталляция — попытка установить соединение, если найдены уязвимые сервисы;

6) получение управления — попытка установить канал связи, если получен доступ к системе;

7) выполнение действий — проведение дальнейших атак с использованием собранной информации.

По совершенным злоумышленником действиям на каждом этапе реализации КА в сети возможно выявить «следы» его присутствия — IoC [20] (рис. 2).

```
nmap -sP 192.168.1.0/24   # Ping-сканирование сети
nmap -sS 192.168.1.1      # TCP SYN-сканирование портов
nmap -O 192.168.1.1       # Определение операционной системы
nmap -sV 192.168.1.1      # Определение версий служб
```

■ ***Рис. 1.*** Основные команды для сканирования сети

■ ***Fig. 1.*** Basic commands for network scanning

Установление IoC позволяет приступить к разработке методики противодействия сетевой разведке объекта КИИ, которая обеспечит раннее обнаружение и предотвращение данной атаки на разных этапах ее реализации.

Разработка методики противодействия сетевой разведке объекта КИИ на основе IoC-анализа

С учетом того, что были установлены IoC, которые появляются в сети на каждом этапе реализации КА, была поставлена задача по разработке методики противодействия сетевой разведке объекта КИИ, которая должна позволить в режиме времени, близком к реальному, проводить анализ трафика, подсчет статистики по портам, детектирование IoC (аномальные TTL, всплески ICMP/SYN-пакетов, повышенный DNS-трафик, повторные попытки аутентификации) и выработку предложений по немедленному реагированию на развитие кризисной ситуации.

Ввиду вышеизложенного для противодействия сетевой разведке объекта КИИ предлагается следующая методика, подробное описание которой приведено на рис. 3–7 в виде алгоритма, ее реализующего.

Этапы выполнения предлагаемой методики.

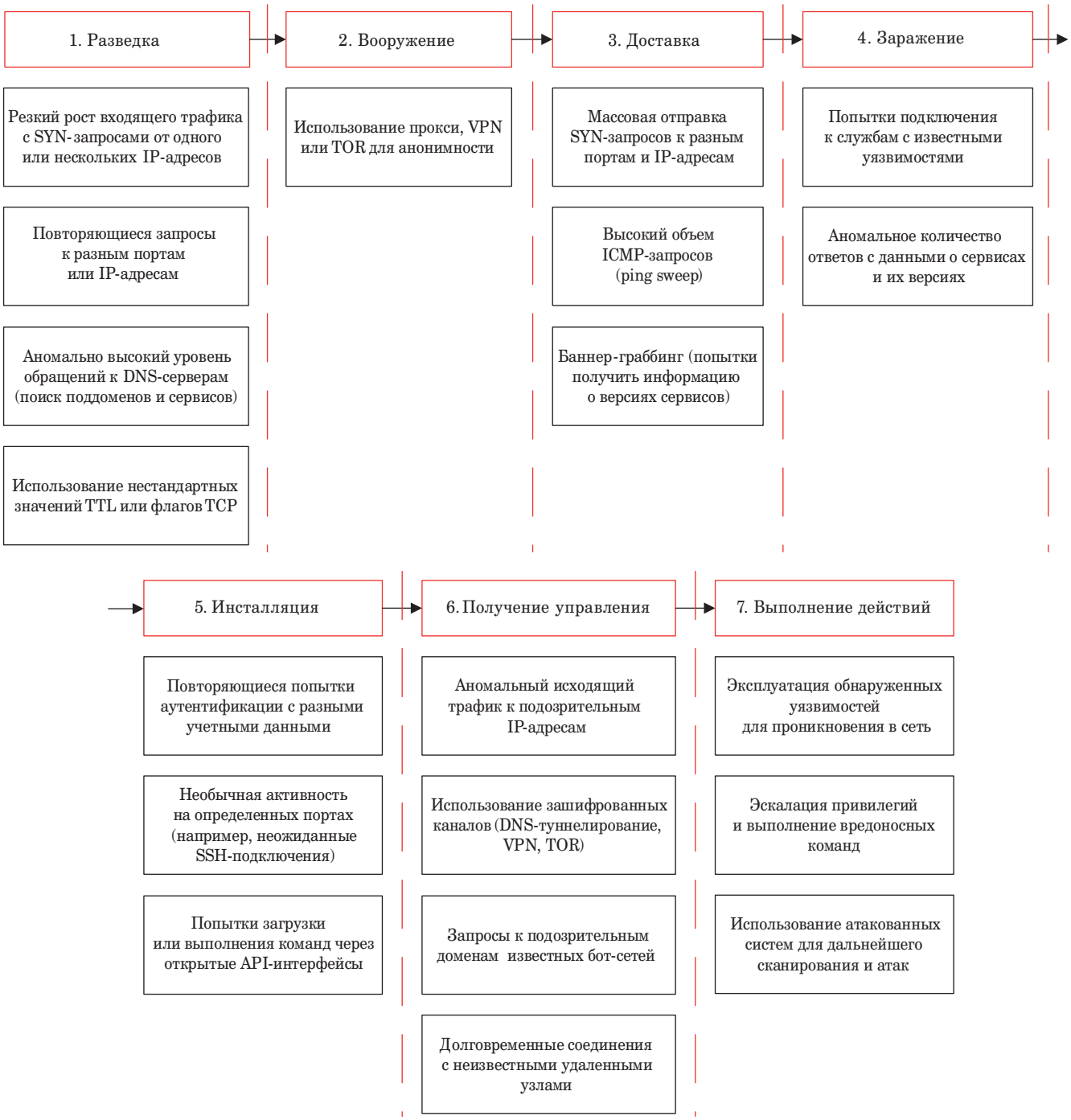
1. Сформировать массивы данных, которые необходимы для работы методики (рис. 3): {*I*} — ресурсы объекта КИИ; {*S*} — средства защиты информации (СЗИ); {*ADM*} – учетные данные о пользователях с правами «Администратор»; {*ADM-Q*} — карантинные данные о пользователях с правами «Администратор»; {*DNS-Q*} — данные, которые не соответствуют записям на сервере DNS; {*X-Q*} — информация об аномальных запросах к удаленным узлам объекта КИИ; {*I-Q*} — информация об удаленных узлах объекта КИИ, помещенных в карантин.

Массивы данных представляют собой наборы информации о ресурсах объекта КИИ, СЗИ и учетных данных пользователей, содержащие такие данные, как внутренний и внешний IP-адреса, маски подсети, MAC-адреса сетевой карты, DNS-серверы, наименование провайдеров и др.

2. Во время информационного обмена провести:

— анализ потоков данных на ресурсах объекта КИИ;

— выявление IoC (аномальные TTL, всплески ICMP/SYN-пакетов), которые соответствуют первому и второму этапам реализации сетевой разведки;



■ ***Рис. 2.*** IoC сетевой разведки объекта КИИ
■ ***Fig. 2.*** IoC network intelligence of a critical infrastructure facility

– устранение уязвимости объекта КИИ, связанной с использованием открытых данных о нем (рис. 4).

3. Если нелегитимная активность на объекте КИИ не прекратилась, то провести:

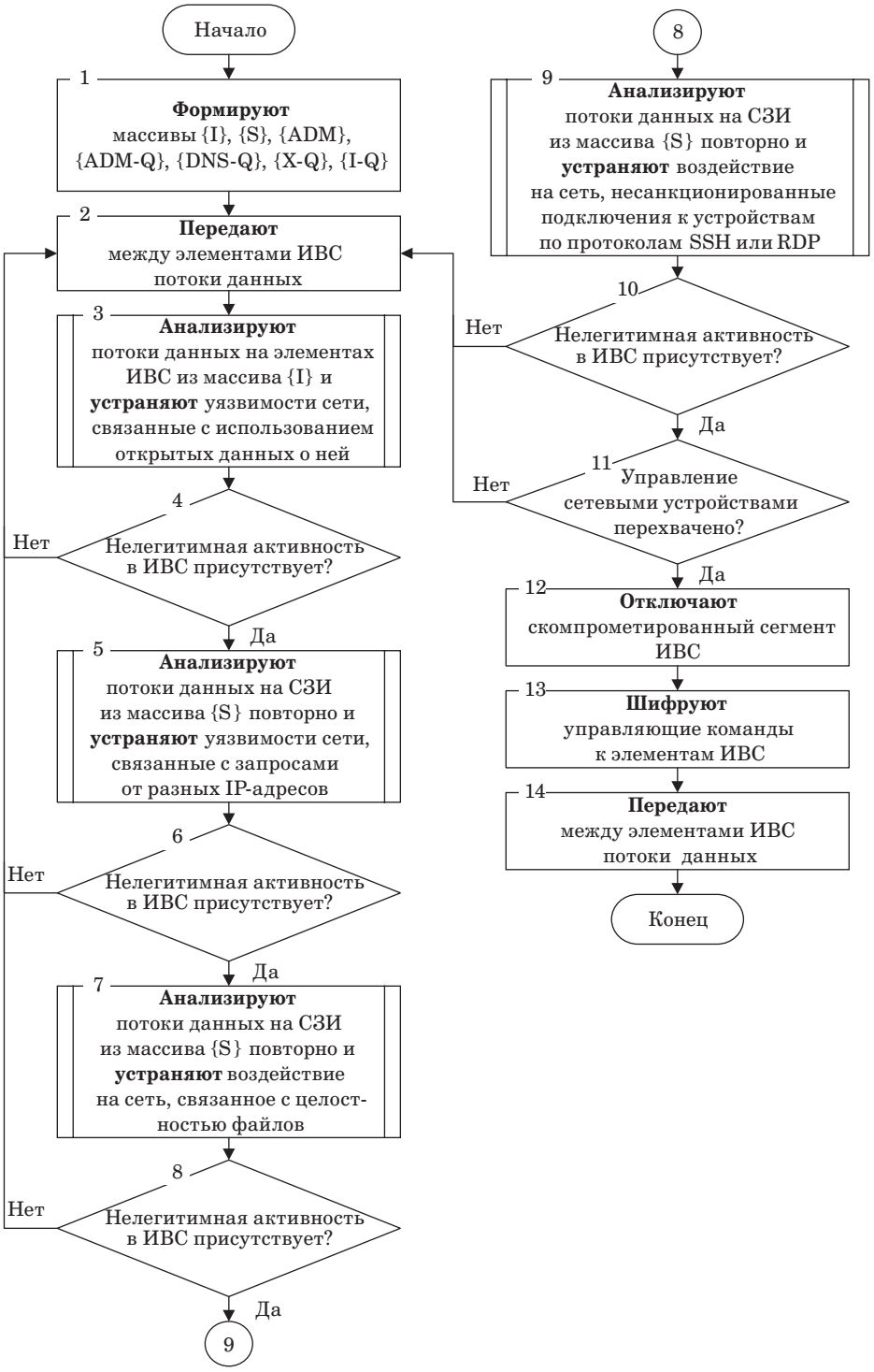
– анализ потоков данных на СЗИ;

– выявление IoC (всплески SYN-пакетов, повышенный DNS-трафик), которые соответствуют третьему и четвертому этапам реализации данной КА;

– устранение уязвимости сети, связанной с запросами от разных IP-адресов (рис. 5).

4. Если нелегитимная активность на объекте КИИ все же не прекратилась, то проводится:

– повторный анализ потоков данных на СЗИ;

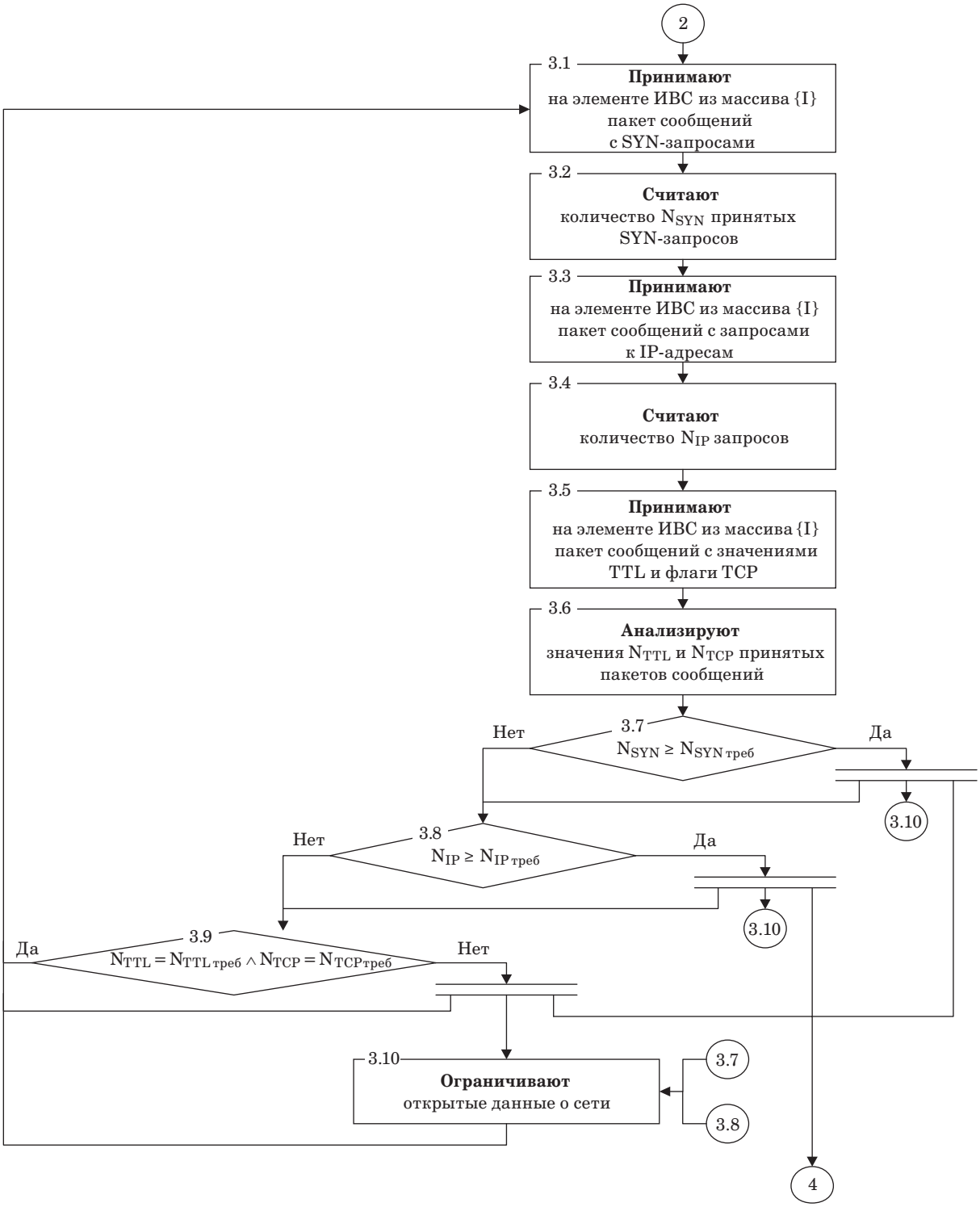


■ ***Рис. 3.*** Блок-схема алгоритма, реализующего методику противодействия сетевой разведке на основе IoC-анализа: *ИВС* — информационно-вычислительная сеть

■ ***Fig. 3.*** Flowchart of an algorithm implementing a method for countering network intelligence based on IoC analysis: *ИВС* — information and computing network

— выявление IoC (изменение учетных данных о пользователях с правами «Администратор», повторные попытки аутентификации, всплески SYN-пакетов, загрузки или выполнения команд через открытые API-интерфейсы), которые соответствуют пятому этапу реализации данной КА;

— устранение воздействия на сеть, связанного с целостностью файлов (рис. 6).



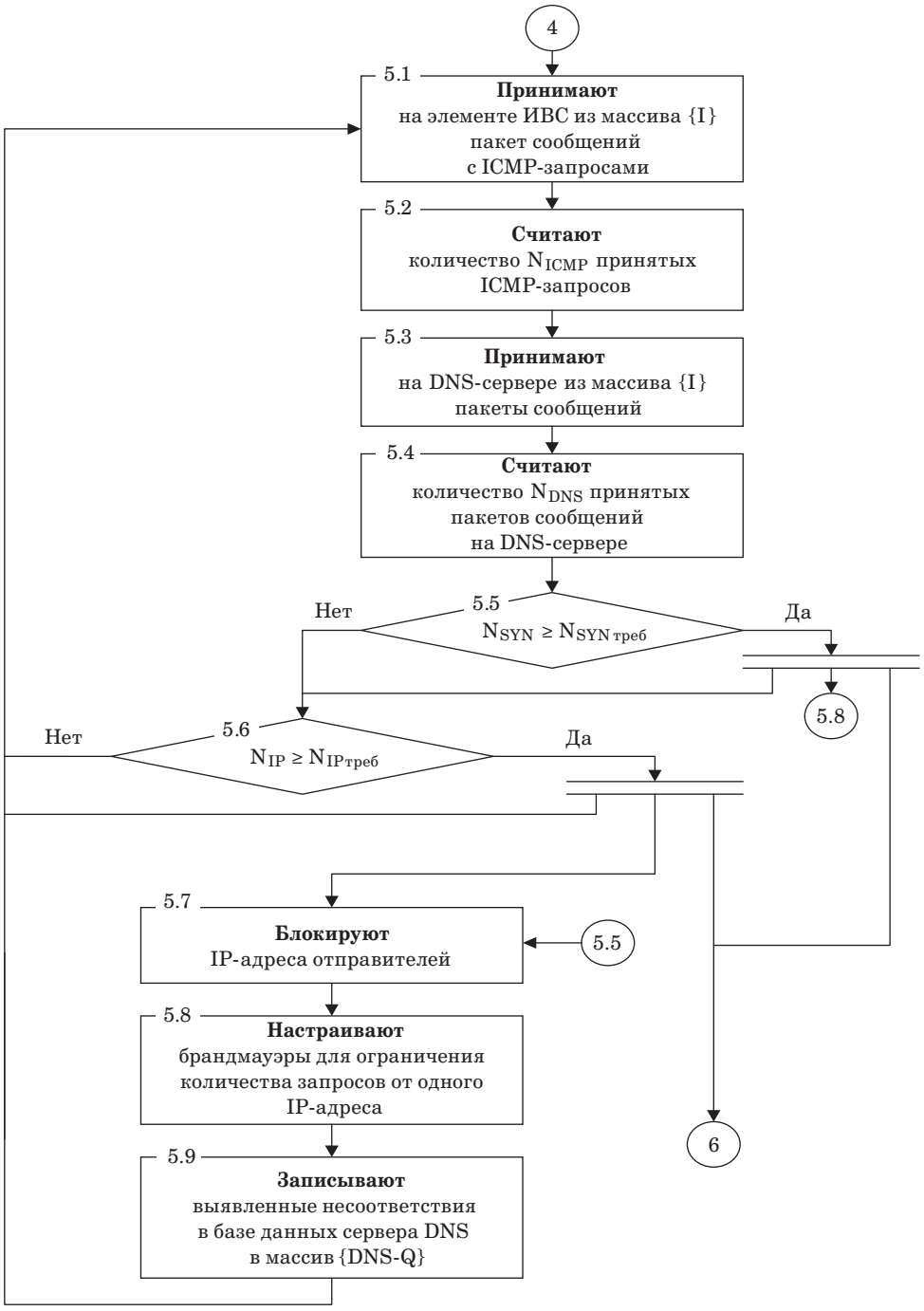
■ ***Рис. 4.*** Блок-схема алгоритма, реализующего методику противодействия сетевой разведке на основе IoC-анализа (*продолжение рис. 3*)

■ ***Fig. 4.*** Flowchart of an algorithm implementing a method for countering network intelligence based on IoC analysis (*continued Fig. 3*)

5. Если нелегитимная активность на объекте КИИ все еще не прекратилась, то проводится:

— повторный анализ потоков данных на СЗИ;

— выявление IoC (аномальные задержки, подозрительные запросы к удаленным узлам объекта КИИ, подключение к карантинным удален-



■ ***Рис. 5.*** Блок-схема алгоритма, реализующего методику противодействия сетевой разведке на основе IoC-анализа (*продолжение рис. 3*)

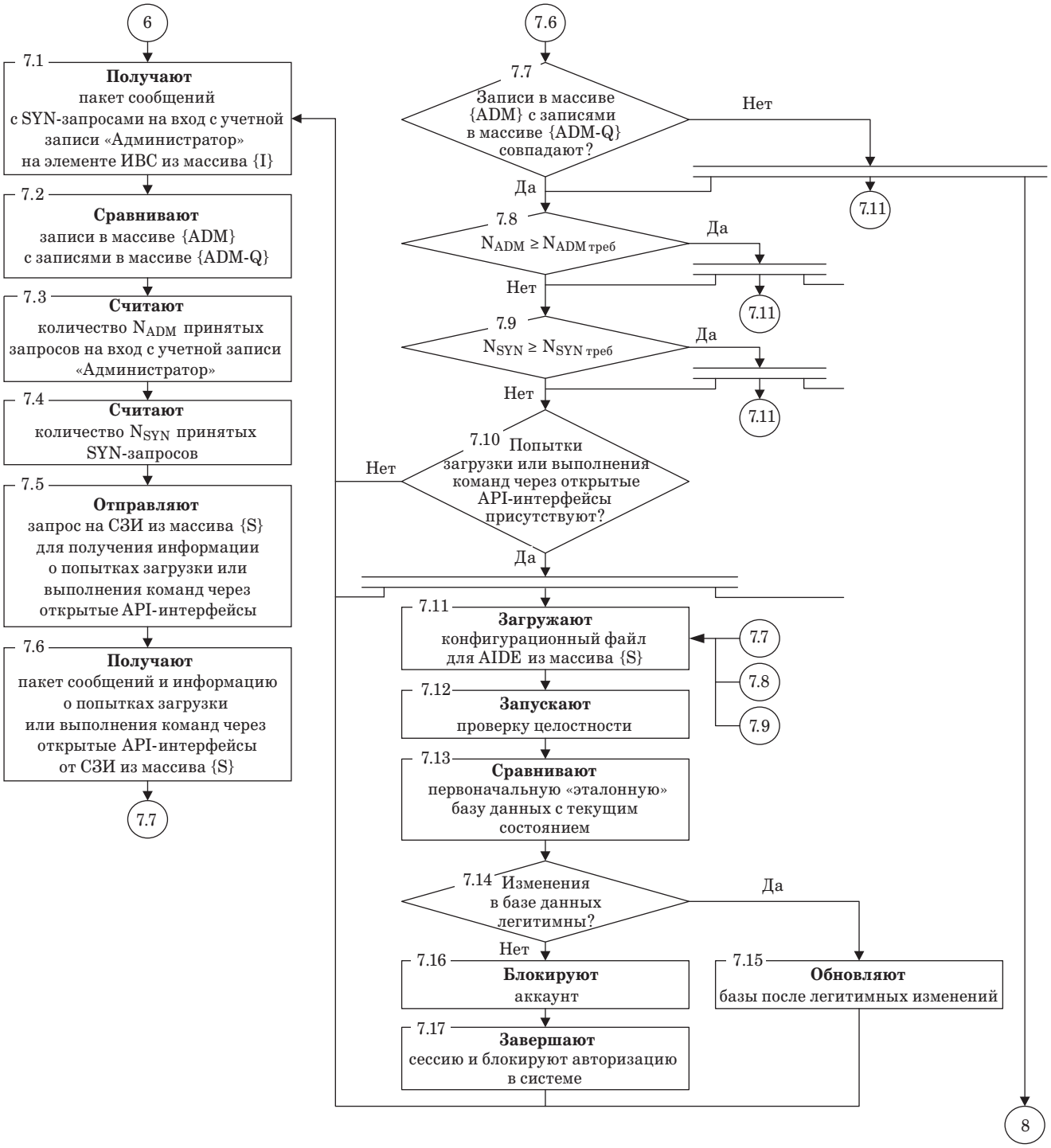
■ ***Fig. 5.*** Flowchart of an algorithm implementing a method for countering network intelligence based on IoC analysis (*continued Fig. 3*)

ным узлам объекта КИИ), которые соответствуют шестому этапу реализации данной КА;

– устранение несанкционированных подключений к устройствам по протоколам SSH или RDP (рис. 7).

6. Если развитие КА находится на финальном этапе, то проводят отключение скомпрометированного сегмента объекта КИИ и осуществляют шифрование управляющих команд (см. рис. 3).

Таким образом, предложена методика противодействия сетевой разведке, учитывающая все этапы эскалации информационного конфликта между злоумышленником и объектом КИИ для незамедлительного его завершения.

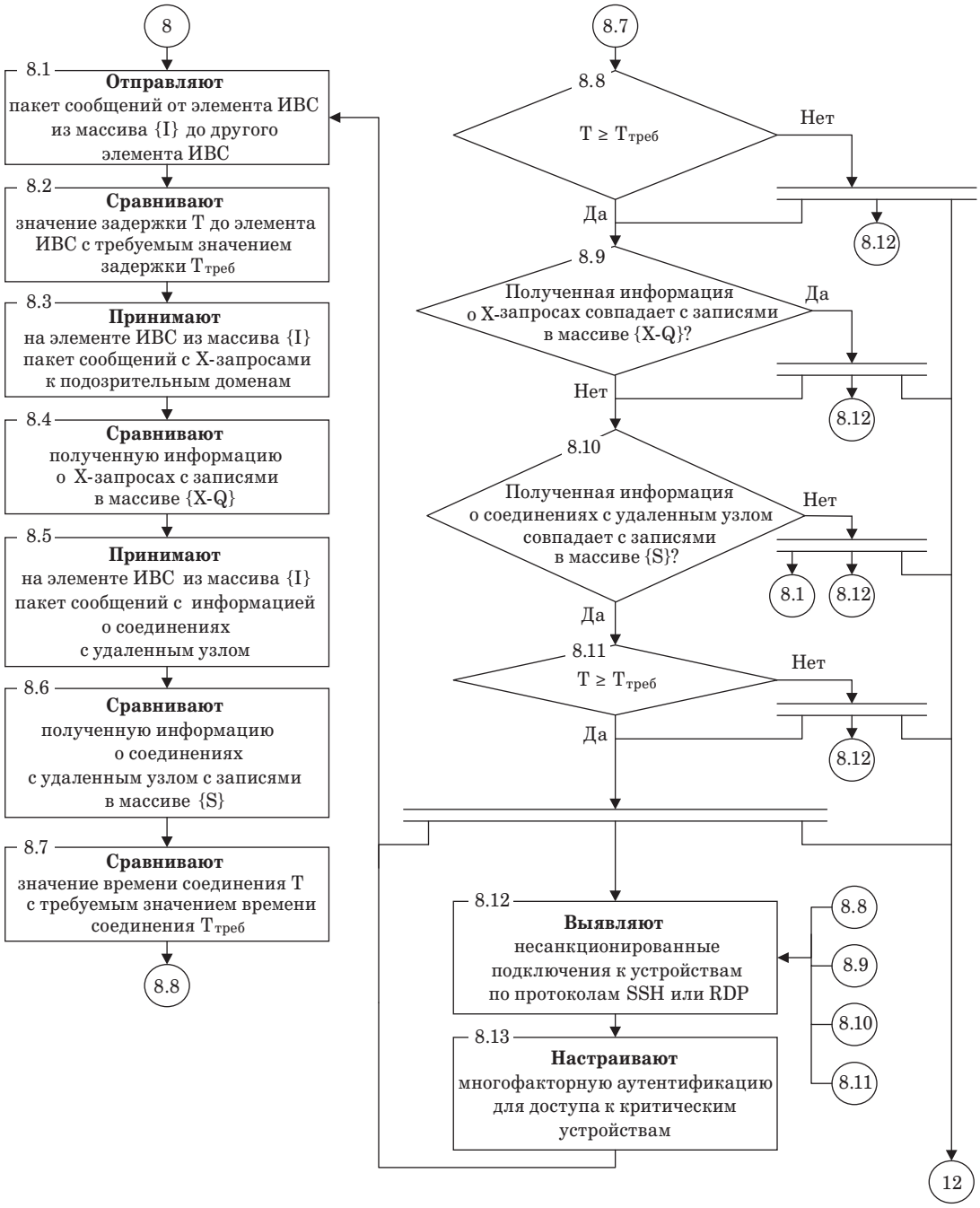


■ ***Рис. 6.*** Блок-схема алгоритма, реализующего методику противодействия сетевой разведке на основе IoC-анализа (*продолжение рис. 3*)

■ ***Fig. 6.*** Flowchart of an algorithm implementing a method for countering network intelligence based on IoC analysis (*continued Fig. 3*)

Реализация разработанной методики в виде ПО

Для того чтобы провести анализ результативности предложенной методики в экспериментах, необходимо разработать ряд инструментальных подходов для автоматизированного выявления IoC и попыток несанкционированного доступа, таких как SYN-запросы, высокий уровень обращения к DNS-записям,



■ ***Рис. 7.*** Блок-схема алгоритма, реализующего методику противодействия сетевой разведке на основе IoC-анализа (*окончание рис. 3*)

■ ***Fig. 7.*** Flowchart of an algorithm implementing a method for countering network intelligence based on IoC analysis (*completion Fig. 3*)

нестандартное значение TTL, высокий объем ICMP-запросов, повторяющиеся попытки аутентификации.

Для отслеживания состояний IoC, связанных с трафиком, необходимо проводить анализ приходящих и исходящих пакетов. В связи с этим для разработки ПО предлагается использовать язык программирования Python и библиотеку Scapy.

Scapy — это мощная интерактивная библиотека на Python для создания, отправки, перехвата и анализа сетевых пакетов. Она сочетает в себе функциональность генератора пакетов, анализатора трафика и декодера протоколов,

при этом предоставляя богатый программный API для автоматизации сетевых экспериментов, тестирования и прототипирования. Scapy давно используется как в исследовании безопасности и «пентестинге», так и в сетевой диагностике и обучении.

Проверка на аномальное поведение TTL происходит путем отправки в функцию самого пакета и массива стандартных значений; если такового нет, то функция передает True, что означает необходимость отправки события на сервер (рис. 8).

Проверка на аномальное количество SYN-запросов происходит путем подсчета количества пакетов с флагом «*S*»; если их количество превышает переданный предел *THRESHOLD* во времени *TIME_WINDOW*, то функция передает True, что означает необходимость отправки события на сервер (рис. 9).

Для того чтобы проверить аномальное количество ICMP/DNS-пакетов, а также трафика на других портах, был создан список экземпляров класса Port, в который записываются значение порта, имя порта и предельное количество проходящих байт в секунду (рис. 10).

Далее на каждый экземпляр класса Port в списке *ports* используется метод этого класса *paket_analyze*(), который содержит набор правил, проверяющих, подходит ли проходящий пакет под характеристики, заданные при инициализации класса, и если подходит, то производится перерасчет среднего значения байт этих пакетов и их схожесть (рис. 11).

Для того чтобы проверить количество попыток входа в систему, необходимо проверить записи во встроенной системе журналирования операционной системы. В различных операционных системах путь к нужному журналу аутентификации разный, например в *Astra Linux* — это */var/log/auth.log* (рис. 12).

Таким образом, было разработано ПО, которое в автоматизированном режиме выявляет IoC и попытки несанкционированного доступа. На рис. 13 представлены результаты регистрации нестандартных значений TTL и высокого объема ICMP-запросов.

```python
from scapy.all import IP

# Определяем стандартное значение TTL

def ttl_check(packet, standart_ttl):
    if IP in packet:
        ttl = packet[IP].ttl
        # Если TTL нестандартный, выводим сообщение
        if ttl not in standart_ttl:
            print(f"Нестандартный ttl: {ttl}")
            return True
    return False
```

■ ***Рис. 8.*** Листинг кода, реализующего выявление аномальных значений TTL

■ ***Fig. 8.*** Listing of code implementing the detection of abnormal TTL values

```python
ports = []

ports.append(Port(23, 'telnet', 25000))
ports.append(Port(22, 'ssh', 60000))
ports.append(Port(21, 'ftp'))
ports.append(Port(80, 'http', 1000))
ports.append(Port(443, 'https', 20000))
ports.append(Port(0, 'icmp', 1000))
ports.append(Port(161,'SNMP'))
ports.append(Port(162,'SNMP'))
ports.append(Port(10161,'SNMP_tls'))
ports.append(Port(10162,'SNMP_tls'))
ports.append(Port(179,'BGP', 100))
ports.append(Port(53, 'DNS', 10000))
```

■ ***Рис. 10.*** Список анализируемых портов

■ ***Fig. 10.*** List of analyzed ports

```python
def syn_check(packet, THRESHOLD, TIME_WINDOW):
    if packet.haslayer(TCP) and packet[TCP].flags == "S":
        src_ip = packet[IP].src
        now = time.time()
        # Добавляем временную метку
        syn_requests[src_ip].append(now)
        # Удаляем старые метки (вне окна TIME_WINDOW)
        syn_requests[src_ip] = [t for t in syn_requests[src_ip] if now - t < TIME_WINDOW]
        if len(syn_requests[src_ip]) > THRESHOLD:
            print(f"[ALERT] SYN flood detected from {src_ip} — {len(syn_requests[src_ip])} SYNs!")
            return True
        else:
            return False
```

■ ***Рис. 9.*** Листинг кода, реализующего проверку на аномальное количество SYN-запросов

■ ***Fig. 9.*** Listing of code implementing a check for an abnormal number of SYN requests

```python
def packet_analyze(self, packet) -> None:
    """Анализирует входящий пакет и решает, учитывать ли его в статистике."""
    self.analyze_counter += 1

    try:
        layers = {layer.name: layer for layer in packet.layers()}

        rules = [
            lambda l: "ICMP" in l and self.port_num == 0,
            lambda l: "TCP" in l and (
                l["TCP"].sport == self.port_num or l["TCP"].dport == self.port_num
            ),
            lambda l: "UDP" in l and (
                l["UDP"].sport == self.port_num or l["UDP"].dport == self.port_num
            ),
            lambda l: "DNSQR" in l,
        ]

        for rule in rules:
            if rule(layers):
                self._handle_packet(packet)
                break

    except Exception as e:
        print(f"[WARN] Ошибка анализа пакета на порту {self.name}: {e}")
```

- ***Рис. 11.*** Листинг кода, реализующего анализ входящих пакетов для подсчета статистики
- ***Fig. 11.*** Listing of code implementing the analysis of incoming packets for calculating statistics

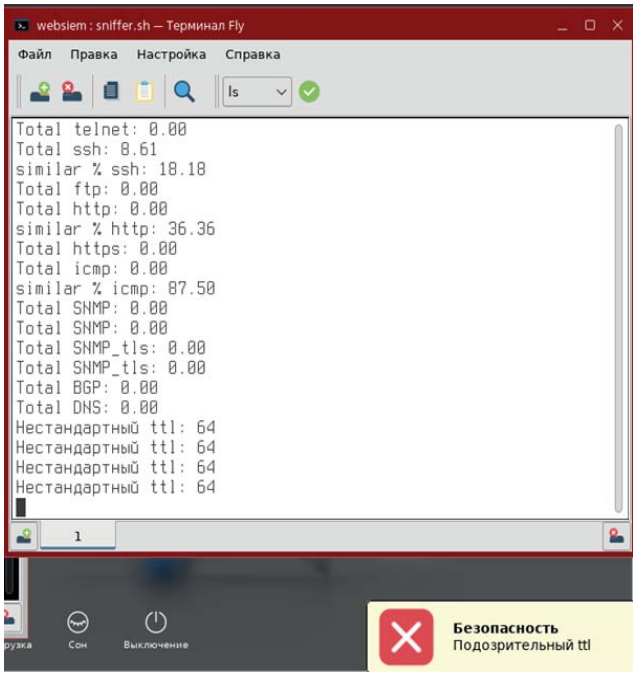
```python
try:
    with open(log_file_path, 'r') as log_file:
        log_file.seek(0, 2)  # Перейти в конец файла
        # old_code = ""
        while True:
            line = log_file.readline()
            if not line:  # Если нет новых строк, подождать 1 секунду
                time.sleep(1)
                continue
            if ('authentication failure' in line):
                # old_code = code
                message = "Найдена ошибка аутентификации: " + line.strip()
                print("Найдена ошибка аутентификации:", line.strip())
                thr = ThreatInfo(message, device='astra', date=datetime.datetime.now(), level=4)
                event["event_type"] = thr.name
                send_event(conf.address, conf.token, event)
                popup_message(message[:90], notif_timer)

except FileNotFoundError:
    print(f"Файл {log_file_path} не найден.")
except PermissionError:
    print(f"Нет доступа к файлу {log_file_path}.")
```

- ***Рис. 12.*** Листинг кода, реализующего проверку журнала на ошибки аутентификации
- ***Fig. 12.*** Listing of code implementing log checking for authentication errors

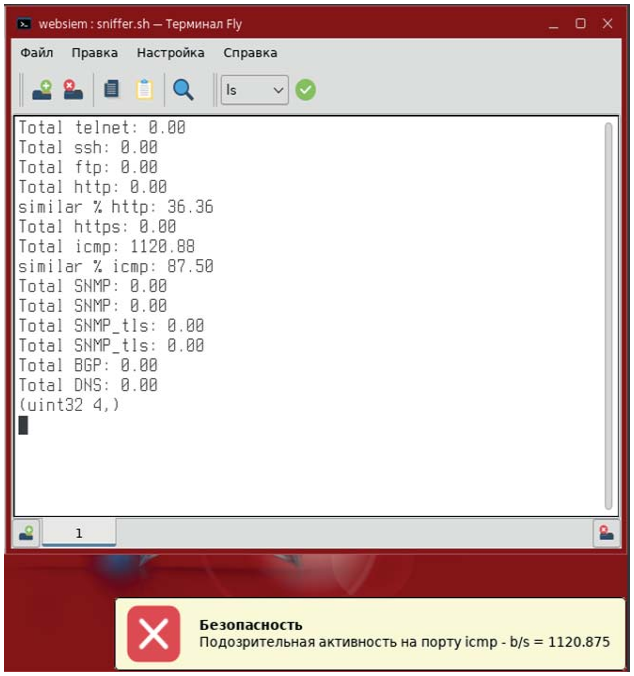
Оценка степени достижения цели

Для оценки степени достижения цели был проведен ряд эксперементальных проверок на лабораторном стенде в виде сети виртуальных машин с одним ядром центрального процессора и 8 ГБ оперативной памяти. В данной виртуальной сети проводилось моделирование сетевой разведки. Анализ трафика проводился с помощью специализированного ПО Wireshark и разработанного ПО на основе предлагаемой методики. Сравнительный анализ времени обнаружения IoC данными средствами представлен в таблице, причем в первом продукте для выяв-

а)



б)



■ ***Рис. 13.*** Результаты экспериментальной проверки разработанного ПО: *а* — регистрация нестандартных значений TTL; *б* — регистрация высокого объема ICMP-запросов

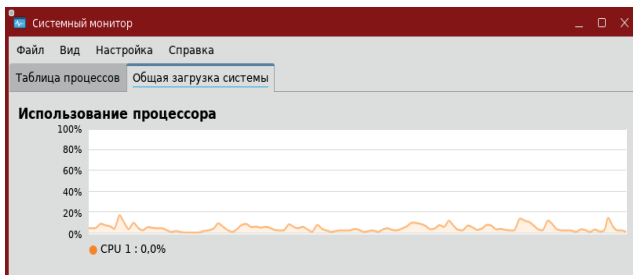
■ ***Fig. 13.*** Results of experimental testing of the developed software: *а* — logging of non-standard TTL values; *б* — logging of high volume of ICMP requests

■ Сравнительный анализ времени обнаружения IoC

■ Comparative analysis of IoC detection times

IoC	Время обнаружения	
	ПО Wireshark	ПО, реализующее предлагаемую методику
Подозрительный TTL	<1 с	<1 с
Подозрительное количество SYN-запросов	>2 с	<1 с
Подозрительный объем трафика на порт	<2 с	<1 с
Одинаковые пакеты	–	<1 с

а)



б)

■ ***Рис. 14.*** Использование ресурсов виртуальной машины во время работы: *а* — ПО Wireshark; *б* — ПО, реализующего предлагаемую методику

■ ***Fig. 14.*** Virtual machine resource usage during operation: *а* — Wireshark software; *б* — software implementing the proposed method

ления IoC (подозрительный TTL, подозрительное количество SYN-запросов, подозрительный объем трафика на порт) необходимо настраивать отдельные фильтры, а во втором продукте данные IoC выявляются в автоматизированном режиме. Данный факт и результаты анализа указывают на то, что при использовании предлагаемой методики оперативность реагирования на угрозы выше, чем при использовании традиционных подходов. Также с помощью ПО Wireshark не представляется возможным провести подсчет одинаковых поступающих сетевых пакетов на объект КИИ в автоматизированном режиме.

Дополнительно проведено сравнение использования ресурсов на виртуальной машине между ПО Wireshark и разработанным ПО. В режиме анализа трафика ПО Wireshark использует от 5 до 20 % ресурса процессора виртуальной машины (рис. 14, *а*), а разработанное ПО в режиме активной обработки и отправки событий использует от 1 до 10 % ресурса процессора (рис. 14, *б*). Ввиду этого можно сделать вывод, что разработанное ПО во время работы задействует в два раза меньше ресурсов виртуальной машины, что позволяет оптимально использовать ограниченный вычислительный ресурс.

Заключение

Представленное исследование демонстрирует, что сетевая разведка — многообразное по методам и целям явление, которое может выступать как инструментом легитимного аудита, так и одним из этапов при проведении многоэтапных атак. Ключевыми задачами являлись исследование методов сетевой разведки и моделирование процесса реализации данной атаки, разработка методики противодействия указанной угрозы объекта КИИ на основе раннего выявления IoC и немедленного реагирования на складывающуюся кризисную ситуацию. Решение данных задач требует сочетания различных техник (ICMP/ARP/TCP/UDP/DNS/SNMP) и специализированных инструментов (*nmap, arp-scan, snmpwalk, DNSenum* и др.). Установлены IoC (аномальные значения TTL, всплески ICMP/SYN-пакетов, повышенные DNS-запросы, повторные попытки аутентификации), которые позволяют формализовать критерии обнаружения риска и служат основой для автоматизированного мониторинга.

Предложенная методика проверена в ряде экспериментов с помощью разработанного на ее основе ПО, которое представляет собой анализатор трафика и коллектора журналов аутентификации с использованием библиотеки Scapy в Python и простой логики пороговых проверок — подтверждает применимость подхода «сбор + корреляция»: анализ сетевых пакетов в сочетании с локальными логами повышает уверенность в выявлении инцидентов и снижает число ложных срабатываний. Использование классов для агрегирования статистики по портам и реализованные правила детектирования (анализ TTL, подсчет SYN/ICMP-пакетов, контроль объема трафика) дают модульную архитектуру, удобную для расширения и интеграции с системами SIEM/IDS.

Научная новизна заключается в комплексном подходе к выявлению сетевой разведки и аномального поведения трафика в сочетании с анализом сетевых пакетов и локальных журналов аутентификации. Предложенная методика детектирования IoC отличается от традиционных методов защиты объектов КИИ тем, что позволяет анализировать трафик, подсчитывать статистику по портам, выявлять аномалии и нейтрализовать КА данного типа в режиме времени, близком к реальному, что повышает оперативность обнаружения угроз.

Теоретическая значимость заключается в развитии методологических положений теории управления ИБ объектов КИИ за счет:

— использования системного подхода для анализа механизмов реализации сетевой разведки и ее моделирования, что позволило установить конкретные IoC, которые возможно зафиксировать в сети на каждом этапе реализации данного воздействия злоумышленника;

— синтеза полученных знаний о IoC и способов обеспечения защищенности сети, что позволило разработать методику противодействия сетевой разведке объекта КИИ.

Практическая значимость методики определяется возможностью ее использования при разработке перспективных информационно-управляющих систем обеспечения ИБ объектов КИИ с применением технологий искусственного интеллекта.

Положительный эффект разработанной методики заключается в том, что ее применение позволяет повысить оперативность обнаружения вероятных угроз объекта КИИ.

Литература

1. *Федеральный закон от 26.07.2017 N 187-ФЗ «О безопасности критической информационной инфраструктуры Российской Федерации»*. https://gossopka.ru/doc/npa/federalnye-zakony/federalnyy-zakon-ot-26072017-n-187-fz-o-bezopasnosti-kriticheskoy-informacionnoy-infrastruktury-ross-37407.html (дата обращения: 10.10.2025).
2. *Методика оценки угроз безопасности информации*. https://fstec.ru/dokumenty/vse-dokumenty/spetsialnye-normativnye-dokumenty/metodicheskij-dokument-ot-5-fevralya-2021-g.html (дата обращения: 05.11.2025).
3. **Липатников В. А., Шевченко А. А., Мелехов К. В., Задбоев В. А.** Метод активной защиты объектов критической информационной инфраструктуры от кибератак на основе прерывания процесса воздействия нарушителя. *Информационно-управляющие системы*, 2025, № 2 (135), с. 37–49. doi:10.31799/1684-8853-2025-2-37-49, EDN: PVFXD
4. **Ajayi O. O., Alozie C. E., Abieba O. A.** Enhancing cybersecurity in energy infrastructure: Strategies for safeguarding critical systems in the digital age. *Trends in Renewable Energy*, 2025, vol. 11, no. 2, pp. 201–212. doi:http://dx.doi.org/10.17737/tre.2025.11.2.00192
5. **Akinbolaji T. J.** Advanced integration of artificial intelligence and machine learning for real-time threat detection in cloud computing environments. *Iconic Research and Engineering Journals*, 2024, vol. 6, no. 10, pp. 980–991. doi:10.5281/zenodo.13963676
6. Пат. 2839562 C1 Российская Федерация, МПК G06F 12/14, H04L 12/22. *Способ защиты информационно-вычислительной сети от вторжения,* В. А. Задбоев, В. А. Липатников, К. В. Мелехов, А. А. Шевченко. № 2024104981; заявл. 27.02.2024; опубл. 06.05.2025.
7. **Shukla P.**, **Krishna C. R., Patil N. V.** Kafka-Shield: Kafka Streams-based distributed detection scheme for IoT traffic-based DDoS attacks. *Security and Privacy*, 2024, vol. 7, iss. 12. doi:10.1002/spy2.416
8. **Ичетовкин Е. А., Котенко И. В.** Модели и алгоритмы защиты систем обнаружения вторжений от атак на компоненты машинного обучения. *Computational nanotechnology*, 2025, т. 12, № 1, с. 17–25. doi:10.33693/2313-223X-2025-12-1-17-25. EDN: LSJCNO
9. **Sarhan M., Layeghy S., Moustafa N., Portmann M.** *Netflow Datasets for Machine Learning-Based Network Intrusion Detection Systems*. In: Big Data Technologies and Applications. Springer International Publishing, 2021, pp. 117–135.
10. **Chen S. S., Hwang R. H., Ali A., Lin Y. D., Wei Y. C., Pai T. W.** Improving quality of indicators of compromise using STIX graphs. *Computers & Security*, 2024, vol. 144, Art. 103972. doi:https://doi.org/10.1016/j.cose.2024.103972
11. **Kartak V., Bashmakov N.** Method for selecting indicators of data compromise. *2022 International Siberian Conference on Control and Communications*, 2022, pp. 1–5. doi:10.1109/SIBCON56144.2022.10002962
12. **Everson D., Cheng L.** A survey on network attack surface mapping. *Digital Threats: Research and Practice*, 2024, vol. 5, no. 2, pp. 1–25. doi:https://doi.org/10.1145/3640019
13. **Tundis A., Modo Nga E. M., Mühlhäuser M.** An exploratory analysis on the impact of Shodan scanning tool on the network attacks. *ARES 2021: The 16th International Conference on Availability, Reliability and Security*, 2021, pp. 1–10. https://doi.org/10.1145/3465481.3469197
14. **Nasereddin M., ALKhamaiseh A., Qasaimeh M.** A systematic review of detection and prevention techniques of SQL injection attacks. *Information Security Journal: A Global Perspective*, 2023, vol. 32, no. 4, pp. 252–265. doi:10.1080/19393555.2021.1995537
15. **Velankar M. R., Mahalle P. N., Shinde G. R.** *Machine Thinking: New Paradigm Shift*. In: Cognitive Computing for Machine Thinking. Innovations in Sustainable Technologies and Computing, 2024. 98 p.
16. **Chernyagin A. S., Svetunkov S. G.** Study of the concept of Bayesian optimization and practical use of its algorithms in the Python programming language. *Technoeconomics*, 2023, vol. 2, no. 4 (7), pp. 4–15. doi:https://doi.org/10.57809/2023.2.4.7.1
17. **Задбоев В. А., Абрамова Н. И., Москалев В. С.** Противодействие внешним вторжениям в информационно-вычислительной сети. *Телекоммуникации и связь*, 2025, № 5 (8), с. 18–23. doi:10.21681/3034-4050-2025-5-18-23, EDN VSBDMK
18. **Липатников В. А., Мелехов К. В., Задбоев В. А.** Способ активной защиты информационно-вычислительных сетей от многоэтапных атак. *Региональная информатика и информационная безопасность: сб. тр. СпБ междунар. конф.*, Санкт-Петербург, 23–25 октября 2024 г. СПб., 2024, с. 112–114.
19. **Sun N., Ding M., Jiang J., Xu W., Mo X., Tai Y.** Cyber threat intelligence mining for proactive cybersecurity defense: A survey and new perspectives. *IEEE Communications Surveys & Tutorials*, 2023, vol. 25, no. 3, pp. 1748–1774. doi:10.1109/COMST.2023.3273282
20. **Parashchuk I., Levshun D., Kotenko I.** Proactive complex security analysis of IoT-based critical infrastructure facilities. *2025 International Russian Smart Industry Conference*, 2025, pp. 52–57. doi:10.1109/SmartIndustryCon65166.2025.10986286

UDC 004.056.53
doi:10.31799/1684-8853-2026-1-61-76
EDN: OIUQDB

Methodology based on the IoC analysis for countering attacker network reconnaissance on a critical information infrastructure facility

A. A. Shevchenko[a], PhD, Tech., Associate Professor, orcid.org/0000-0001-9113-1089
V. A. Lipatnikov[b], Dr. Sc., Tech., Professor, orcid.org/0000-0002-3736-4743, lipatnikovanl@mail.ru
V. A. Zadboev[b], Junior Researcher, orcid.org/0009-0003-9362-1307
P. I. Kuzin[c], PhD, Tech., Associate Professor, orcid.org/0000-0003-0880-6204
[a]The Bonch-Bruevich Saint-Petersburg State University of Telecommunications, 22-1, Bolshevikov Pr., 193232, Saint-Petersburg, Russian Federation
[b]S. M. Budenny Military Academy of Communication, 3, Tikhoretskii Pr., 190064, Saint-Petersburg, Russian Federation
[c]Saint Petersburg State Forest Technical University named after S. M. Kirov, 5, Institutsky per., 194021, Saint-Petersburg, Russian Federation

Introduction: The spread of new ways to target critical information infrastructure facilities developed and used by attackers encourages the search for relevant countermeasures. **Purpose:** Analyzing various ways of network reconnaissance on critical information infrastructure facilities, to develop a methodology based on the IoC analysis for countering this type of attacker influence in order to improve the efficiency of detecting potential threats to a critical information infrastructure facility. **Results:** Using a system approach, we analyze network reconnaissance mechanisms (ICMP, TCP/UDP, ARP, DNS, SNMP) and tools (nmap, arp-scan, DNSenum, snmpwalk) for detecting and identifying active devices, open ports, operating systems, and services. On the basis of this analysis, we develop a model for this type of attack, which allows identifying specific IoCs that can be recorded in the network at each stage of an attacker's invasion. Synthesis of the acquired knowledge about key IoCs, such as abnormal TTL values, bursts of ICMP and SYN packets, increased DNS traffic, and repeated authentication attempts, with network security methods enables developing a methodology based on the IoC analysis for countering network reconnaissance on a critical information infrastructure facility, and a number of instrumental approaches for the rapid detection of network traffic anomalies and unauthorized access attempts using the Scapy Python library. The implementation of individual stages of the proposed methodology in the form of software has facilitated the analysis of its effectiveness in a number of experiments. **Practical relevance:** The proposed methodology can be used for developing information management systems which ensure the information security of critical information infrastructure facilities.

Keywords — information security, critical information infrastructure facility, unauthorized access, traffic analysis, IoC, model, early threat mitigation.

For citation: Shevchenko A. A., Lipatnikov V. A., Zadboev V. A., Kuzin P. I. Methodology based on the IoC analysis for countering attacker network reconnaissance on a critical information infrastructure facility. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2026, no. 1, pp. 61–76 (In Russian). doi:10.31799/1684-8853-2026-1-61-76, EDN: OIUQDB

References

1. *Federal'nyj zakon ot 26.07.2017 N 187-FZ "O bezopasnosti kriticheskoj informacionnoj infrastruktury Rossijskoj Federacii"* [Federal Law of 26.07.2017 N 187-FZ "On the Security of Critical Information Infrastructure of the Russian Federation"]. Available at: https://gossopka.ru/doc/npa/federalnye-zakony/federalnyy-zakon-ot-26072017-n-187-fz-o-bezopasnosti-kriticheskoy-informacionnoy-infrastruktury-ross-37407.html (accessed 10 October 2025).
2. *Metodika ocenki ugroz bezopasnosti informacii* [Information security threats assessment methodology]. Available at: https://fstec.ru/dokumenty/vse-dokumenty/spetsialnye-normativnye-dokumenty/metodicheskij-dokument-ot-5-fevralya-2021-g.html (accessed 5 November 2025).
3. Lipatnikov V. A., Shevchenko A. A., Melekhov K. V., Zadboev V. A. Active protection method against cyberattacks for the objects of critical information infrastructure based on the interruption of the process of an intruder's impact. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 2, pp. 37–49. doi:10.31799/1684-8853-2025-2-37-49, EDN: PVFXD.
4. Ajayi O. O., Alozie C. E., Abieba O. A. Enhancing cybersecurity in energy infrastructure: strategies for safeguarding critical systems in the digital age. *Trends in Renewable Energy*, 2025, vol. 11, no. 2, pp. 201–212. doi:http://dx.doi.org/10.17737/tre.2025.11.2.00192.
5. Akinbolaji T. J. Advanced integration of artificial intelligence and machine learning for real-time threat detection in cloud computing environments. *Iconic Research and Engineering Journals*, 2024, vol. 6, no. 10, pp. 980–991. doi:10.5281/zenodo.13963676.
6. Zadboev V. A., et al. *Sposob zashchity informacionno-vychislitel'noj seti ot vtorzheniya* [A method for protecting an information and computing network from intrusion]. Patent Russian Federation, no. 2024104981, 2025.
7. Shukla P., Krishna C. R., Patil N. V., Kafka-Shield: Kafka Streams-based distributed detection scheme for IoT traffic-based DDoS attacks. *Security and Privacy*, 2024, vol. 7, iss. 12. doi:10.1002/spy2.416
8. Ichetovkin E. A., Kotenko I. V. Models and algorithms for protecting intrusion detection systems from attacks on machine learning components. *Computational nanotechnology*, 2025, vol. 12, no. 1, pp. 17–25 (In Russian). doi:10.33693/2313-223X-2025-12-1-17-25. EDN: LSJCNO
9. Sarhan M., Layeghy S., Moustafa N., Portmann M. *Netflow Datasets for Machine Learning-Based Network Intrusion Detection Systems*. In: *Big Data Technologies and Applications*. Springer International Publishing, 2021, pp. 117–135.
10. Chen S. S., Hwang R. H., Ali A., Lin Y. D., Wei Y. C., Pai T. W. Improving quality of indicators of compromise using STIX graphs. *Computers & Security*, 2024, vol. 144, Art. 103972. doi:https://doi.org/10.1016/j.cose.2024.103972
11. Kartak V., Bashmakov N. Method for selecting indicators of data compromise. *2022 International Siberian Conference on Control and Communications*, 2022, pp. 1–5. doi:10.1109/SIBCON56144.2022.10002962
12. Everson D., Cheng L. A survey on network attack surface mapping. *Digital Threats: Research and Practice*, 2024, vol. 5, no. 2, pp. 1–25. doi:https://doi.org/10.1145/3640019
13. Tundis A., Modo Nga E. M., Mühlhäuser M. An exploratory analysis on the impact of Shodan scanning tool on the network attacks. *ARES 2021: The 16th International Conference on Availability, Reliability and Security*, 2021, pp. 1–10. https://doi.org/10.1145/3465481.3469197
14. Nasereddin M., ALKhamaiseh A., Qasaimeh M. A systematic review of detection and prevention techniques of SQL injection attacks. *Information Security Journal: A Global Perspective*, 2023, vol. 32, no. 4, pp. 252–265. doi:10.1080/19393555.2021.1995537
15. Velankar M. R., Mahalle P. N., Shinde G. R. *Machine Thinking: New Paradigm Shift*. In: *Cognitive Computing for Machine Thinking*. Innovations in Sustainable Technologies and Computing, 2024. 98 p.
16. Chernyagin A. S., Svetunkov S. G. Study of the concept of Bayesian optimization and practical use of its algorithms in the Python programming language. *Technoeconomics*, 2023,

vol. 2, no. 4 (7), pp. 4–15. doi:https://doi.org/10.57809/2023.2.4.7.1
17. Zadboev V. A., Abramova N. I., Moskalev V. S. Countering external intrusions into the information and computing network. *Telecommunications and Communications*, 2025, no. 5 (8), pp. 18–23 (In Russian). doi:10.21681/3034-4050-2025-5-18-23, EDN VSBDMK
18. Lipatnikov V. A., Melekhov K. V., Zadboev V. A. A method for actively protecting information and computing networks from multi-stage attacks. *Sbornik trudov Sankt-Peterburgskoj mezhdunarodnoj konferencii "Regional'naya informatika i informacionnaya bezopasnost"* [Proceedings of the St. Petersburg International Conference "Regional informatics and information security"]. Saint Petersburg, 2024, pp. 112–114 (In Russian).
19. Sun N., Ding M., Jiang J., Xu W., Mo X., Tai Y. Cyber threat intelligence mining for proactive cybersecurity defense: A survey and new perspectives. *IEEE Communications Surveys & Tutorials*, 2023, vol. 25, no. 3, pp. 1748–1774. doi:10.1109/COMST.2023.3273282
20. Parashchuk I., Levshun D., Kotenko I. Proactive complex security analysis of IoT-based critical infrastructure facilities. *2025 International Russian Smart Industry Conference*, 2025, pp. 52–57. doi:10.1109/SmartIndustryCon65166.2025.10986286

ПАМЯТКА ДЛЯ АВТОРОВ

Поступающие в редакцию статьи проходят обязательное рецензирование.

При наличии положительной рецензии статья рассматривается редакционной коллегией. Принятая в печать статья направляется автору для согласования редакторских правок. После согласования автор представляет в редакцию окончательный вариант текста статьи.

Процедуры согласования текста статьи могут осуществляться как непосредственно в редакции, так и по e-mail (ius.spb@gmail.com).

При отклонении статьи редакция представляет автору мотивированное заключение и рецензию, при необходимости доработать статью — рецензию.

Редакция журнала напоминает, что ответственность за достоверность и точность рекламных материалов несут рекламодатели.

УДК 004.89
doi:10.31799/1684-8853-2026-1-77-88
EDN: MNREWK

Научные статьи
Articles

Критерии выбора модели трансферного обучения для решения задач анализа изображений

Т. Э. Шульга[a], *доктор физ.-мат. наук, профессор, orcid.org/0000-0002-5521-5960, taiss@yandex.ru*
А. А. Сытник[a], *доктор техн. наук, профессор, orcid.org/0000-0002-1256-7253, as@sstu.ru*
Д. А. Солопекин[a], *аспирант, orcid.org/0009-0006-7546-0150*
[a]*Саратовский государственный технический университет им. Гагарина Ю. А., Политехническая ул., 77, Саратов, 410054, РФ*

Введение: *выбор исходной модели для задач трансферного обучения в области анализа изображений представляет собой серьезную проблему, несмотря на высокую эффективность данного подхода. Отсутствие систематизированной процедуры на этапе выбора начальной архитектуры ограничивает качество и применимость конечных решений, особенно при работе с дефицитными вычислительными ресурсами и узкоспециализированными прикладными задачами. Кроме того, в настоящее время отсутствует унифицированный набор критериев, позволяющих проводить объективную оценку и сравнение различных предобученных моделей.* ***Цель:*** *разработать и апробировать структурированную систему критериев для выбора исходной модели трансферного обучения в задачах анализа изображений.* ***Результаты:*** *проведен комплексный сравнительный анализ современных подходов к трансферному обучению. В рамках исследования определены пять ключевых критериев, оптимизирующих процесс выбора исходной модели. В качестве экспериментального подтверждения предложенные критерии были применены для решения задачи сегментации пешеходов и транспортных средств на городских изображениях. В результате использования предложенного подхода была получена модель с метриками: loss = 0,24, mean F1 = 0,78, mean IoU = 0,7. Данные результаты сопоставимы с лучшими типовыми решениями при значительно меньших затратах времени и вычислительных ресурсов.* ***Практическая значимость:*** *предложенная система критериев выбора исходной модели может существенно повысить эффективность и воспроизводимость внедрения трансферного обучения в прикладных задачах компьютерного зрения, включая такие области, как медицинская диагностика, автономный транспорт и др.* ***Обсуждение:*** *перспективным направлением дальнейших исследований является формализация предложенных критериев в виде унифицированной методологии для обеспечения большего удобства и простоты их практического применения.*

Ключевые слова *— трансферное обучение, преимущества и недостатки моделей трансферного обучения, глубокое обучение, критерии выбора моделей, компьютерное зрение.*

Для цитирования: Шульга Т. Э., Сытник А. А., Солопекин Д. А. Критерии выбора модели трансферного обучения для решения задач анализа изображений. *Информационно-управляющие системы*, 2026, № 1, с. 77–88. doi:10.31799/1684-8853-2026-1-77-88, EDN: MNREWK

For citation: Shulga T. E., Sytnik A. A., Solopekin D. A. Criteria for selecting a transfer learning model for image analysis tasks. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2026, no. 1, pp. 77–88 (In Russian). doi:10.31799/1684-8853-2026-1-77-88, EDN: MNREWK

Введение

Анализ изображений представляет собой ключевую исследовательскую задачу в области компьютерного зрения. Она заключается в разработке эффективных алгоритмов и методов для автоматического извлечения информации из цифровых изображений, включая видеопоследовательности. Основная сложность анализа обусловлена значительным разнообразием и сложностью структуры изображений, их изменчивостью, а также влиянием таких факторов, как шум и изменяющиеся условия освещения, что затрудняет точное извлечение и интерпретацию данных. К основным задачам анализа изображений относятся: автоматическое обнаружение объектов, их классификация, сегментация и распознавание образов. Наиболее точными, быстрыми и устойчивыми системами для обработки изображений на сегодня считаются решения, основанные на методах нейронных сетей и глубокого обучения.

Среди прикладных задач, представляющих наибольший интерес, можно выделить несколько преобладающих направлений: анализ медицинских изображений [1], геоданных [2]; распознавание дорожных знаков [3, 4], транспортных средств, техники и живых организмов.

Этап создания модели для решения прикладных задач распознавания, сегментации или классификации является одним из наиболее продолжительных и трудоемких в цикле разработки. Его сложность обусловлена, во-первых, сложностью архитектур нейронных сетей, а во-вторых, значительными аппаратными ресурсами, требуемыми для обучения модели. Минимизировать эти трудности позволяет подход трансферного обучения, который применяется в различных об-

ластях знаний с использованием существующих предобученных моделей [5].

Одной из ключевых проблем данного подхода является отсутствие четких критериев выбора исходной предобученной сети, служащей источником знаний. Это подтверждается исследователями, которые признают, что выбор подходящей исходной модели представляет собой новую сложную задачу в рамках трансферного обучения [6].

Целью данной работы является формулировка набора критериев, позволяющих повысить эффективность выбора исходной сети. Это в свою очередь позволит достичь максимальной результативности обучения при сокращении требуемых временных и аппаратных ресурсов.

Для достижения поставленной цели необходимо проанализировать существующие преимущества и недостатки моделей трансферного обучения, а также формализовать проблему выбора исходной сети. Решение этих задач даст возможность систематизировать полученные знания и определить комплекс критериев выбора.

Преимущества и особенности использования моделей трансферного обучения

Трансферное обучение позволяет использовать знания, полученные при решении одной задачи, для повышения эффективности решения другой, часто родственной задачи. Этот подход особенно ценен в условиях ограниченного объема размеченных данных для целевой задачи, поскольку дает возможность использовать предобученные модели для ускорения и оптимизации процесса обучения.

Важным этапом работы с трансферными моделями является дообучение на целевом наборе данных. Этот процесс заключается в адаптации модели, изначально обученной на обширном и разнородном наборе данных, к конкретной задаче с помощью небольшого специализированного набора данных. Стратегия дообучения часто включает заморозку начальных слоев модели, которые содержат общие признаки, и обучение только конечных слоев, ответственных за специфические для задачи характеристики [7]. Такой подход позволяет сохранить полезные обобщенные знания модели и значительно снижает риск переобучения.

Эффективность различных архитектур нейронных сетей для задач компьютерного зрения может существенно различаться.

Классификация изображений. Согласно анализу литературы для решения задач классификации широко применяются сверточные нейронные сети (Convolutional Neural Networks, CNN) [8, 9], такие как VGG, ResNet, Inception и EfficientNet. Их архитектура включает множество сверточных и пулинговых слоев. Ключевое преимущество использования предобученных CNN — возможность дообучения модели, где замороженные слои служат экстракторами универсальных признаков. Это делает CNN предпочтительным выбором для работы с малыми размеченными наборами данных. Эффективность различных моделей и степень заморозки их слоев напрямую зависят от степени близости (схожести) новой задачи и исходной задачи, на которой модель была предобучена (адаптация домена [10]).

Распознавание объектов. Эта задача сложнее классификации, так как требует не только определить категорию объекта, но и локализовать его на изображении с помощью ограничивающих рамок (bounding boxes). Для ее решения применяются такие архитектуры, как YOLO [11], SSD [12] и Faster R-CNN [13]. Следует выделить семейство моделей YOLO, которые демонстрируют высокую эффективность в сценариях с необходимостью обнаружения множества объектов в реальном времени благодаря высокой скорости обработки и сохранению хорошей точности.

Семантическая сегментация. Это одна из наиболее сложных задач, заключающаяся в присвоении каждому пикселю изображения метки класса, что позволяет точно определять границы объектов. Этот метод широко востребован в таких областях, как медицинская визуализация для выделения патологических образований (например, опухолей на рентгенологических снимках [14]). Среди наиболее распространенных архитектур для сегментации можно выделить U-Net [15], SegNet и Mask R-CNN. Архитектуры U-Net и Mask R-CNN особенно хорошо подходят для задач, требующих высокой точности и детализации разметки.

Таким образом, для каждой из рассмотренных задач был определен набор архитектур, демонстрирующих наилучшие результаты. Однако, несмотря на различия, модели, основанные на этих архитектурах, имеют ряд общих недостатков.

Существующие проблемы в применении моделей трансферного обучения для решения практических задач

В ходе работы с моделями трансферного обучения был выявлен ряд ключевых проблем: трудность адаптации к специфическим данным, сильная зависимость от объема данных, высокие требования к вычислительным ресурсам, огра-

ниченная интерпретируемость результатов, а также сложность выбора исходной модели.

1. Трудность адаптации к специфическим данным. Трансферные модели могут демонстрировать низкую эффективность при переносе знаний на сильно отличающиеся (по домену) наборы данных. Чем специфичнее целевые данные, тем хуже способность модели к адаптации. Как отмечено в исследовании [16], данную проблему можно частично нивелировать путем тонкой настройки (тюнинга) модели. При этом общепризнанно, что высокая схожесть между исходным и целевым доменами напрямую коррелирует с лучшими результатами.

2. Зависимость от объема данных. Проблема зависимости эффективности обучения от размера выборки является фундаментальной для глубокого обучения. Считается, что для успешного дообучения трансферных моделей необходим достаточно большой объем размеченных данных. Это подтверждается исследованием британских ученых [17], в котором анализировались девять различных трансфертных сетей, имеющих четыре различных типа архитектуры (DenseNet, Inception-v3, ResNet, VGG) на трех наборах данных изображений геологических материалов разного размера (7000, 41 812 и 104 306 изображений).

Была выявлена явная положительная корреляция между объемом данных и итоговой точностью модели. При этом разные архитектуры показывали наилучшие результаты на наборах разного размера. Например, Inception-v3 показала наивысшую точность на крупнейшем наборе данных (104 306 изображений), в то время как VGG19 лидировала на самом маленьком наборе (7000 изображений), но ее результаты улучшались не так значительно с ростом данных. Исследователи также отметили, что модели, обученные на малых выборках, сильно склонны к переобучению. Для борьбы с этим эффектом рекомендуется оставлять больше замороженных слоев. Для больших наборов данных, напротив, можно размораживать и обучать больше слоев, что часто приводит к лучшим результатам. Дополнительным методом борьбы с переобучением и искусственного увеличения данных является аугментация.

3. Высокие требования к вычислительным ресурсам. Обучение и применение трансферных моделей, особенно на больших наборах данных, предъявляет значительные требования к вычислительным мощностям. В исследовании [17] отмечается, что корреляция между ростом производительности и вычислительной сложностью при больших ресурсах составляет 0,635 (средняя степень влияния, что указывает на наличие других значимых факторов).

4. Ограниченная интерпретируемость результатов. Глубокие нейронные сети, включая трансферные модели, часто работают как «черный ящик», что затрудняет интерпретацию их решений и является критическим ограничением в таких областях, как медицина [18]. Исследователи описывают попытки решения этой проблемы как «вскрытие черного ящика». В настоящее время не достигнут консенсус относительно универсальной системы интерпретации. В работе [18] предлагается использовать четыре типа объяснений в зависимости от уровня интерпретируемости (глобальный или локальный): объяснение правилами, скрытой семантикой, атрибуцией и на примерах. Однако каждая из этих категорий сама по себе требует сложной интерпретации, так как описывается с помощью трудных для восприятия формальных конструкций.

5. Проблемы с генерализацией. Модели трансферного обучения могут демонстрировать существенное падение производительности на новых данных (out-of-distribution, OOD), которые значительно отличаются от обучающей выборки. Это особенно критично в динамично меняющихся и узкоспециализированных областях. В рамках парадигмы OOD [19] ведутся исследования, направленные на разработку механизмов, позволяющих моделям адекватно работать с данными за пределами исходного распределения.

Таким образом, различные типы задач компьютерного зрения требуют разных архитектур и стратегий трансферного обучения. Для достижения максимальной эффективности при выборе предобученной модели необходимо учитывать все обозначенные выше проблемы. Выбор оптимальной исходной сети на текущий момент остается сложной задачей, не имеющей универсального решения.

Проблема выбора моделей трансферного обучения

Анализ концепции трансферного обучения, его базовых принципов и сопутствующих проблем позволяет заключить, что проблема выбора исходной предобученной модели была изначально неизбежна.

Относительная простота создания и использования предобученных моделей по сравнению с разработкой нейронных сетей «с нуля» привела к появлению множества разнообразных архитектур, демонстрирующих различную эффективность в решении специфических задач. Эти модели активно публиковались энтузиастами в открытом доступе, способствуя популяризации направления и стимулируя его развитие. Как

признали исследователи в 2021 г., широкая доступность предобученных моделей стала одним из краеугольных камней в прогрессе глубокого обучения [20].

Развитие инструментария для создания моделей, эволюция алгоритмов и стремление к универсальности привели к росту как количества доступных моделей, так и сфер их применения. Это сопровождалось увеличением их ресурсоемкости (при недостаточном уровне оптимизации), усложнением методов тонкой настройки и расширением применимости. Со временем рост числа подобных критериев существенно усложнил выбор оптимальной исходной предобученной модели для решения конкретной задачи.

Сформировалось множество парадигм трансферного обучения, ориентированных на различные критерии оценки и построения итоговой модели. Проведенный анализ научной литературы позволил выделить наиболее распространенные парадигмы (табл. 1).

Каждая парадигма предполагает свой, зачастую уникальный и слабо пересекающийся с другими, метод выбора исходной модели. Эти методы не унифицированы и варьируются в зависимости от подхода конкретного исследователя.

Наиболее простым алгоритмом выбора сети является метод грубой силы (brute-force), который заключается в последовательном переборе моделей из определенного репозитория (хаба). При использовании этого метода, даже при сужении выборки для сокращения времени работы, происходят значительный перерасход вычислительных ресурсов и временные затраты. Результаты сравнения моделей после обучения могут быть визуализированы, например, в виде графика (рис. 1) [21].

Таким образом, некорректный подход к выбору модели трансферного обучения ведет к негативным последствиям: бесполезной трате времени, перерасходу вычислительных ресурсов (и, как следствие, финансовым потерям при использовании облачной инфраструктуры или износу собственного оборудования), а также к получению менее эффективной итоговой модели.

Можно заключить, что разработка системного набора критериев для обоснованного выбора модели позволит минимизировать указанные риски и предложить решение проблемы выбора в трансферном обучении.

Описание критериев выбора модели

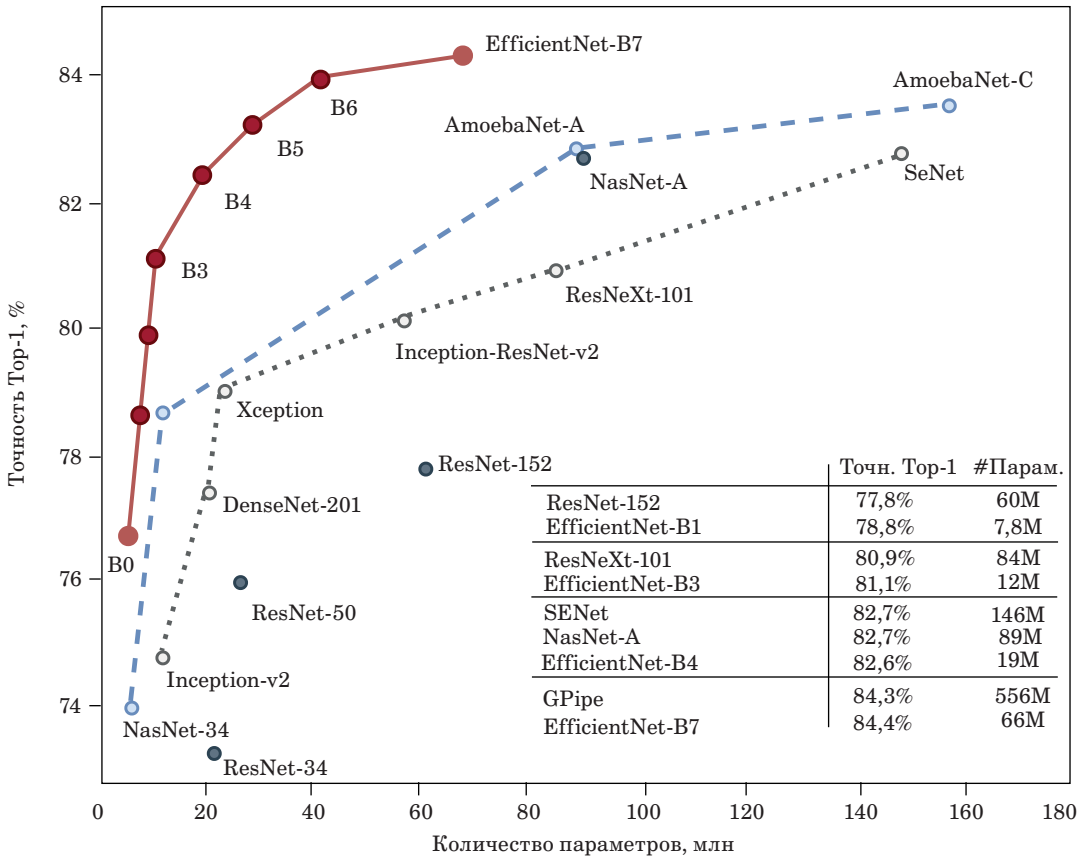
В качестве первого критерия для выбора моделей предлагается **доступность**: модель должна находиться в открытом доступе (например, в репозиториях типа Hugging Face Hub или TensorFlow Hub), чтобы обеспечить возможность ее загрузки, использования и дообучения.

Вторым важнейшим критерием является **размер модели**: количество параметров должно соответствовать доступным вычислительным ресурсам и требованиям к времени инференса/обучения.

Анализ соревновательной статистики репозиториев предобученных моделей позволяет сформулировать простой первичный принцип выбора: отдать предпочтение модели с наивысшей

■ ***Таблица 1.*** Парадигмы обучения моделей трансферного обучения
■ ***Table 1.*** Transfer learning model training paradigms

Парадигма обучения	Входные данные	Критерий сравнения
Оценка переносимости модели, не зависящей от источника	Репозиторий (хаб) обученных моделей, целевой набор данных	Точность на целевой задаче
Оценка переносимости модели, зависящей от источника	Репозиторий (хаб) обученных моделей, исходный набор данных, целевой набор данных	Точность на целевой задаче
Оценка переносимости задач	Репозиторий (хаб) исходных задач, целевой набор данных	Точность на целевой задаче
Предсказание способности к генерализации	Обученная модель, тестовый набор данных	Разница между точностью на тестовом и обучающем наборе
Предсказание ошибки OOD	Обученная модель, обучающая выборка, OOD тестовый набор	Погрешность на тестовом наборе
Валидация при обучении с учителем	Ключевые точки процесса обучения, валидационный набор данных	Точность на тестовом наборе
Валидация при обучении без учителя	Ключевые точки процесса обучения, тестовый набор данных	Точность тестового набора



■ ***Рис. 1.*** Оценка качества модели в зависимости от числа параметров
■ ***Fig. 1.*** Model quality evaluation graph depending on the number of parameters

точностью Top-1 Accuracy (Top-1 Acc). Однако следует учитывать, что этот показатель может варьироваться в зависимости от тестовой выборки. Например, на момент написания работы лидером по метрике Top-1 Acc на наборе данных ImageNet-1k (https://paperswithcode.com/sota/image-classification-on-imagenet;) является модель CoCa [22], которая демонстрирует результат 0,91 в режиме тонкой настройки (fine-tuning) и 0,9 в режиме замороженных слоев (frozen).

Большое количество параметров напрямую влияет на потребление ресурсов оперативной (RAM) и видеопамяти (VRAM), а также на требования к вычислительным мощностям (графическим (GPU) или тензорным (TPU) процессорам). Это увеличивает стоимость обучения и эксплуатации модели, что противоречит одной из целей трансферного обучения — экономии ресурсов.

В контексте трансферного обучения количество параметров определяет объем знаний, которые модель уже содержит. Как правило, более крупные сети обладают большей способностью к генерализации и могут демонстрировать более высокую эффективность на новых задачах (как с дообучением, так и без него).

Для сравнения потребления вычислительных моделей проведен системный анализ, позволивший определить основные критерии сравнения: значения минимального объема VRAM, задержка прямого распространения и пропускная способность. Приведенные диапазоны основаны на сочетании теоретических оценок объема памяти, требуемой для хранения весов и промежуточных активаций (https://developer.nvidia.com/blog/gpu-memory-essentials-for-ai-performance/; https://shayangeek.com/en/estimate-gpu-memory-needed-for-ai-model/; https://huggingface.co/blog/train_memory;), и опубликованных бенчмарков быстродействия типовых моделей на различных GPU (https://www.exxactcorp.com/blog/Benchmarks/nvidia-a100-deep-learning-benchmarks-for-tensorflow; https://github.com/Oneflow-Inc/DLPerf;). Результаты анализа представлены в табл. 2.

Таким образом, наглядно показано, что переход от моделей, которые помещаются в память одного ускорителя с 16–24 ГБ VRAM, к моделям масштаба 1–2 млрд параметров требует либо существенно более емких GPU, либо использования систем с несколькими GPU. Это приводит

■ ***Таблица 2.*** Типизированная ресурсоемкость и быстродействие моделей компьютерного зрения в зависимости от числа параметров

■ ***Table 2.*** Typical resource intensity and performance of computer vision models depending on the number of parameters

Параметры модели, млн	Типичные архитектуры	Типичные задачи	Минимальный объем VRAM для обучения (320×320, batch = 4), ГБ	Задержка прямого распространения на RTX3090 (320×320, batch = 1), мс	Пропускная способность на RTX3090 (320×320, batch = 1), кадр/с
5–20	Легкие сверточные модели: MobileNetV2, EfficientNet-B0, облегченные U-Net/DeepLab	Классификация, простая сегментация, распознавание в реальном времени	4–6	1–5	~200–1000
20–100	ResNet-34/50, U-Net/FPN/LinkNet	Сегментация и детекция общего назначения	6–10	5–15	~70–200
100–300	EfficientNet-B3/B5/B6, ResNeXt-101, гибриды CNN–Transformer	Высокоточная сегментация и детекция	10–18	15–40	~25–70
300–1000	Крупные CV-модели и Vision Transformer: ViT-Large, большие Swin/CoAtNet и др.	Высокоточная аналитика, офлайн-обработка	18–32	40–100	~10–25
>1000	Очень крупные мультимодальные модели: CoCa, CLIP-подобные модели масштаба 1–2 млрд параметров и выше	Сложные мультимодальные задачи	>32	>100	<10

к увеличению времени задержки и снижению пропускной способности, что является основным аргументом в пользу выбора более компактных моделей при ограниченных вычислительных ресурсах.

Даже после сортировки моделей по количеству параметров и метрике Top-1 Acc выборка может оставаться чрезмерно большой (только для задач классификации на ImageNet имеются тысячи общедоступных моделей). Следовательно, критерия размера модели недостаточно.

Третий критерий — **сходство доменов**: исходный домен предобучения модели должен быть релевантен целевой задаче. Название набора данных, на котором тестировалась модель (например, ImageNet), указывает на домен ее предварительного обучения. Наиболее известные наборы данных включают ImageNet, COCO, Pascal VOC, Open Images, Cityscapes, ADE20K, KITTI, MNIST и CIFAR.

Эффективность модели при тонкой настройке напрямую зависит от близости исходного домена к целевой задаче. Модель, предобученная на данных общего характера (ImageNet), будет хорошо работать в задачах классификации и распознавания объектов из ее исходного домена. Однако для высокоспециализированных областей (например, медицинской визуализации) ее эффективность может снижаться. Чем ближе данные целевой задачи к данным предобучения, тем выше потенциал для успешной передачи знаний и адаптации модели.

Четвертый ключевой критерий — **архитектура модели**, которая должна подходить для решения целевой задачи. Внутренняя организация и принципы работы модели определяют ее применимость для конкретных задач. Архитектуры, такие как ResNet, EfficientNet, VGG или трансформеры, обладают различными характеристиками, подходящими для разных сценариев.

Для задач с ограниченными вычислительными ресурсами предпочтительны более легкие архитектуры, такие как MobileNet или EfficientNet. Трансформерные архитектуры, например Vision Transformer (ViT), получили популярность благодаря своей универсальности, особенно в задачах, требующих работы с контекстно-зависимыми признаками.

Пятый критерий — **адаптивность**: модель должна демонстрировать гибкость при дообучении и сохранять производительность на данных, отличающихся от обучающей выборки (в рамках целевого домена). Адаптивность подразумевает также возможность выборочного замораживa-

ния/дообучения слоев и простоту интеграции в существующий рабочий процесс.

Модели с модульной структурой (например, ResNet) позволяют легко замораживать базовые слои и дообучать верхние. Пригодность модели для различных метрик и типов задач (наличие специализированных выходных слоев для классификации, детекции, сегментации) значительно сокращает затраты на ее адаптацию.

Предполагается, что выбор модели может быть осуществлен на основе данных критериев в следующем порядке.

1. Выбрать из списка моделей-кандидатов модели, соответствующие критерию доступности.

2. Из выбранных на предыдущем этапе моделей параллельно определить модели, соответствующие критериям размера, схожести доменов и архитектуре и провести агрегацию результатов (например, одновременное превышение пороговых значений по всем критериям или использование некоторой взвешенной интегральной оценки).

3. Из выделенных на предыдущем этапе моделей выбрать модели, соответствующие критерию адаптивности.

Предложенный подход к выбору модели может претендовать на универсальность. В научной литературе также существуют более узкоспециализированные подходы. Например, в работе [23] предлагается математический алгоритм на основе кривых AUC для каждого независимого признака в целях построения общего индекса разделимости и выбора оптимальной модели для задач классификации.

Пример применения критериев выбора модели

В рамках апробации предложенного подхода выбора исходной трансферной модели на основе описанных критериев была поставлена задача выбора модели для семантической сегментации пешеходов и автомобилей на изображениях.

Набор данных. Для обучения и валидации моделей был выбран набор данных Cambridge-driving Labeled Video Database (CamVid) [24]. Набор состоит из 701 изображения с разрешением 960×720 пикселей, снятых камерой видеорегистратора. Данные аннотированы вручную с разметкой 32 классов объектов. Набор был разделен на обучающую (367 изображений), валидационную (101 изображение) и тестовую (233 изображения) выборки.

Для данного набора данных известно лучшее достигнутое значение метрики усредненного IoU (Intersection over Union), составляющее 0,846. Этот результат был получен на модели SERNet-Former [25], имеющей 44 млн параметров.

Исходные модели для решения задачи семантической сегментации пешеходов и автомобилей на изображениях выбирались на основе пяти предложенных критериев.

Доступность. С учетом критерия доступности были отобраны следующие семейства моделей: EfficientNet, ResNet и Inception.

Сходство доменов. На основе требований задачи исходный домен модели должен включать знания о пешеходах и транспортных средствах. Наиболее релевантными наборами данных были признаны COCO и ImageNet.

Выбор архитектуры. Поскольку целевая задача относится к семантической сегментации, были выбраны четыре популярные для этого класса задач архитектуры: U-Net, FPN, LinkNet и PSPNet. В ходе предварительных экспериментов от архитектуры PSPNet пришлось отказаться из-за высокой вычислительной сложности и длительного времени обучения (архитектура добавляет 8–12 млн параметров к базовой модели).

Размер модели. Для минимизации ресурсных затрат были рассмотрены модели с количеством параметров менее 50 млн: EfficientNet-B3, EfficientNet-B6, ResNet-34, ResNeXt-101, ResNet-152 и Inception-V3.

Рассчитанный итоговый размер моделей для каждой архитектуры представлен в табл. 3.

Адаптивность. Способность модели к адаптации оценивалась в процессе обучения путем сравнения метрик на валидационной выборке после непродолжительного обучения.

В связи с небольшим объемом данных для повышения их вариативности были применены техники аугментации.

Аугментация. Для аугментации обучающей выборки применялся набор искусственных преобразований: случайное отражение, масштабирование до ±50 % и сдвиг до 10 % с добавлением полей и случайным обрезанием до 320×320, добавление гауссова шума и перспективных искажений, а также три группы фотометрических изменений (контраст и яркость, резкость, цвет);

■ ***Таблица 3.*** Размер итоговых моделей, млн параметров

■ ***Table 3.*** Final model sizes, million parameters

Модель\Архитектура	Unet	FPN	LinkNet
ResNet-34	~ 27–32	~ 25–27	~ 24–25
ResNeXt-101	~ 50–55	~ 47–49	~ 45–46
ResNet-152	~ 65–70	~ 63–65	~ 62–63
EfficientNet-B3	~ 17–22	~ 15–17	~ 13–14
EfficientNet-B6	~ 48–53	~ 46–48	~ 45–46
Inception-V3	~ 29–34	~ 27–29	~ 26–27

после этого маски округлялись и ограничивались в [0; 1], чтобы устранить артефакты интерполяции и сохранить строго двоичный формат разметки.

Таким образом, обучающая выборка подвергалась агрессивным и контролируемым преобразованиям, обеспечивающим значительное разнообразие вариантов одной и той же сцены. Это позволило снизить риск переобучения при ограниченном размере исходного набора данных.

Для валидационной выборки применялась только минимальная геометрическая обработка для обеспечения совместимости с требованиями сверточных архитектур по кратности размеров входа и исключения влияния случайных аугментаций на процесс оценки качества.

Экспериментальная установка. Обучение каждой конфигурации модели проводилось в течение 40 эпох с фиксированным размером батча 8. В качестве функции потерь использовалась комбинация кросс-энтропии и Dice-loss. Оптимизация параметров осуществлялась методом оптимизатора Adam с начальной скоростью обучения 0,0001 и стандартным расписанием ее снижения. В качестве функции активации использовалась сигмоида.

Все вычислительные эксперименты выполнялись на персональном компьютере с использованием аппаратного ускорения CUDA. Аппаратная конфигурация включала видеокарту NVIDIA GeForce RTX 3090 с объемом видеопамяти 24 ГБ, 32 ГБ оперативной памяти и процессор AMD Ryzen 9 5900X. Реализация экспериментов была выполнена на языке Python с использованием фреймворка глубокого обучения. Все модели обучались на одной видеокарте. Подготовленная экспериментальная установка отражает типичный сценарий применения трансферного обучения в условиях ограниченных вычислительных ресурсов.

Метрики оценки качества модели. В задачах семантической сегментации модель выполняет покомпонентную классификацию пикселей. Соответственно, для оценки качества используются метрики, основанные на количестве правильно и ошибочно классифицированных пикселей. В работе используются следующие основные метрики:

— значение функции потерь (loss), используемой при обучении;

— усредненная по классам F1-мера (mean F1);

— усредненный по классам показатель пересечения-объединения (mean IoU, Intersection over Union).

Функция потерь отражает степень расхождения между предсказанными вероятностями и целевой разметкой и минимизируется в процессе обучения.

Для каждого класса c на уровне пикселей можно вычислить количество истинно положительных (TP), ложноположительных (FP) и ложноотрицательных (FN) случаев. На их основе определяются:

— IoU для класса c как доля пересечения предсказанной и истинной областей класса по отношению к их объединению:

$$IoU_c = \frac{TP_c}{TP_c + FP_c + FN_c};$$

— F1-мера для класса c как гармоническое среднее точности и полноты:

$$F1_c = \frac{2TP_c}{2TP_c + FP_c + FN_c}.$$

В работе используются усредненные по классам значения этих метрик (mean IoU и mean F1), что позволяет всем классам вносить сопоставимый вклад в итоговую оценку, даже при наличии дисбаланса между ними.

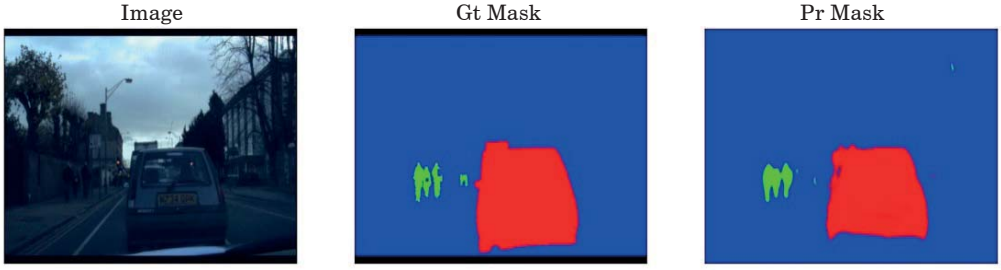
F1-мера использовалась в качестве основной метрики для ранжирования моделей, поскольку она лучше отражает качество сегментации критически важных классов в условиях сильного дисбаланса по сравнению, например, с точностью (accuracy).

Обучение. Были обучены следующие комбинации архитектура — модель: U-Net + EfficientNet-B6, FPN + EfficientNet-B6, LinkNet + ResNet-34, LinkNet + ResNeXt-101, U-Net + ResNet-152, U-Net + Inception-V3, U-Net + EfficientNet-B3 (с улучшенной аугментацией). Выполнялось пять последовательных прогонов для каждого случая, для всех прогонов сформулировано среднее ± стандартное отклонение. Также во время прогонов замерялось потребление памяти, по итогу прогонов подготовлена интегральная оценка потребления памяти VRAM по времени (площадь графика), измеренного в гигабайт-час. Результаты представлены в табл. 4.

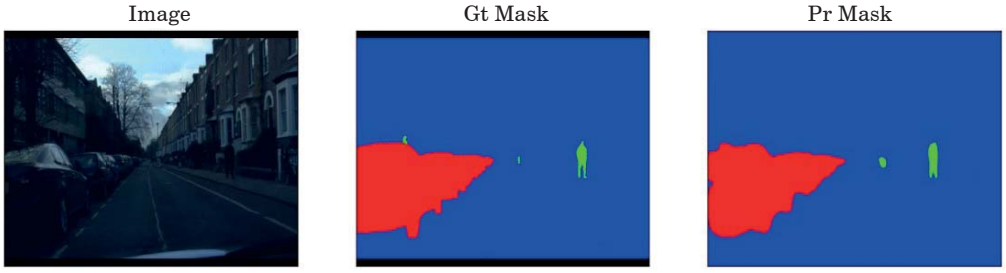
Наилучший результат по всем метрикам был достигнут при строгом следовании предложенному подходу (комбинация U-Net с EfficientNet-B3, рис. 2). Единственным отступлением стала дополнительная настройка аугментации в процессе обучения (снижение вероятностей для уменьшения агрессивности подхода). Без этого шага лучшей оказалась комбинация U-Net с EfficientNet-B6 (рис. 3), которая, уступая по количеству параметров, потенциально могла бы показать более высокие результаты с улучшенной аугментацией. Однако с точки зрения баланса «производительность — ресурсозатраты» комбинация U-Net с EfficientNet-B3

■ ***Таблица 4.*** Результаты обучения комбинаций моделей
■ ***Table 4.*** Results of training model combinations

Архитектура	Модель	Loss ± std	Mean F1 ± std	Mean IoU ± std	Mean, ГБ·ч
unet	efficientnetb6	**0,238±0,019**	**0,773±0,016**	**0,689±0,012**	**3,05**
unet	efficientnetb3	0,259±0,027	0,771±0,014	0,688±0,016	1,72
unet	resnet152	0,274±0,037	0,752±0,029	0,668±0,031	2,09
unet	inceptionv3	0,303±0,001	0,744±0,002	0,664±0,001	1,53
FPN	efficientnetb6	0,268±0,021	0,739±0,016	0,632±0,011	2,94
linknet	resnet34	0,350±0,035	0,674±0,014	0,579±0,015	2,31
linknet	resnext101	0,317±0,064	0,712±0,026	0,612±0,023	3,82
Доработка аугментации					
unet	efficientnetb3	**0,234±0,024**	**0,784±0,015**	**0,694±0,016**	1,69



■ ***Рис. 2.*** Выходные данные модели, обученной с EfficientNetB3 на архитектуре U-Net с доработкой аугментации
■ ***Fig. 2.*** Output data of the model trained with EfficientNetB3 on the U-Net architecture with augmented improvements



■ ***Рис. 3.*** Выходные данные модели, обученной с EfficientNetB6 на архитектуре U-Net
■ ***Fig. 3.*** Output data of the model trained with EfficientNetB6 on the U-Net architecture

является однозначно предпочтительной (экономия составляет порядка 1,36 ГБ·ч на один запуск). Также, в сравнении с последовательными прогонами без следования представленному подходу, определено, что итоговая экономия составила порядка 62,28 ГБ·ч, при суммарном потреблении 95,75 ГБ·ч за все прогоны.

Если бы на шаге выбора начального домена был выбран набор данных COCO, можно было бы добиться и лучших результатов, так как модели выше, исходя из предсказаний, испытывают явные трудности с сегментацией пешеходов. Однако, исходя из критерия доступности, найти такие модели в открытом доступе не удалось.

Заключение

В работе предложен систематизированный подход к выбору исходной предобученной модели для задач трансферного обучения в компьютерном зрении. Подход основан на пяти критериях: доступности, размере модели, доменной близости, архитектуре и адаптивности, — что позволяет преобразовать эмпирический выбор модели в воспроизводимую процедуру, учитывающую как ресурсные ограничения, так и специфику целевой задачи. Дополнительно выполнен обзор типичных диапазонов числа параметров моделей и соответствующих им требований к видеопамя-

ти, времени задержки и пропускной способности на референсной платформе (GPU RTX 3090). Это обосновывает практическую целесообразность применения сравнительно компактных моделей.

Экспериментальная апробация на задаче семантической сегментации для пешеходов и автомобилей на изображениях набора данных CamVid показала, что последовательное применение критериев позволяет существенно сузить пространство перебора моделей и снизить совокупные вычислительные затраты по сравнению с наивным полным перебором. Наилучшее соотношение качества и ресурсопотребления продемонстрировали конфигурации на основе U-Net с энкодерами EfficientNet-B3/B6 и расширенной схемой аугментации. При этом EfficientNet-B3 обеспечивает сопоставимые значения loss, mean F1 и mean IoU при значительно меньшем потреблении ресурсов, чем более тяжелая EfficientNet-B6 (экономия составляет в среднем 1,36 Гб·ч на один запуск).

Для оценки воспроизводимости результатов было выполнено несколько независимых запусков обучения. Итоговые метрики представлены в формате «среднее ± стандартное отклонение». Для лучших конфигураций стандартные отклонения F1-меры и mean IoU не превышают нескольких процентных пунктов, что свидетельствует о стабильности полученных результатов и устойчивости предложенного подхода к вариациям инициализации и стохастичности процесса обучения.

Основным ограничением работы является апробация предложенного подхода исключительно на задаче семантической сегментации в одной предметной области с использованием единственного датасета. Другим ограничением служит относительно простая формализация многокритериального выбора (пороговые правила и эмпирическое ранжирование).

Эти ограничения определяют перспективные направления дальнейших исследований:

— расширение экспериментальной базы за счет применения предложенного подхода к выбору моделей для других типов задач компьютерного зрения (классификация, детекция, мультимодальные сценарии);

— проведение строгого анализа чувствительности результатов к стратегиям аугментации данных и размеру выборки в рамках каждой из задач;

— формализация предложенных критериев в виде интегральной функции полезности, пригодной для автоматизации выбора моделей в репозиториях предобученных сетей.

Литература

1. **Акутин А. С., Горякин М. В., Зубавленко Р. А., Печенкин В. В., Солопекин Д. А.** Использование искусственных нейронных сетей для выполнения сегментации рентгенограмм тазобедренного сустава при лечении остеоартрита. *Моделирование, оптимизация и информационные технологии*, 2024, т. 12, № 1. https://moitvivt.ru/ru/journal/pdf?id=1486 (дата обращения: 24.12.2024). doi:10.26102/2310-6018/2024.44.1.011
2. **Nunez A., Misra S., Falola Y.** Transfer learning for geological carbon storage forecasting using neural operator. *Advances in Water Resources*, 2025, vol. 199. doi:10.1016/j.advwatres.2025.104948
3. **Thanh V. N., Thang P. C., Truong V. V., Thao H. V., Quan T. M., Minh H. N. N.** Traffic sign recognition through the use of an Internet of Things system and deep learning. *Информационно-управляющие системы*, 2025, № 2, с. 2–15. doi:10.31799/1684-8853-2025-2-2-15, EDN: MLUUSQ
4. **Шульга Т. Э., Солопекин Д. А.** Распознавание дорожных знаков российского образца с использованием нейронных сетей. *Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика*, 2024, № 2, с. 85–94. doi:10.24143/2072-9502-2024-2-85-94
5. **Aaryan P., Damodar P., Shaswata M., Sudip M., Shahram R.** Transfer learning applied to computer vision problems: survey on current progress, limitations, and opportunities. *Computer Vision and Pattern Recognition*, 2024, vol. 1, iss. 1, 16 p.
6. **Xue Y., Yang R., Chen X., Liu W., Wang Z., Liu X.** A review on transferability estimation in deep transfer learning. *IEEE Transactions on Artificial Intelligence*, 2024, pp. 1–24. doi:10.1109/TAI.2024.3445892
7. **Zhuang F., Qi Z., Duan K., Xi D., Zhu Y., Zhu H., Xiong H., He Q.** A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 2021, vol. 109, no. 1, pp. 43–76. doi:10.1109/JPROC.2020.3004555
8. **Ramprasath M., Anand M. V., Hariharan S.** Image classification using convolutional neural networks. *International Journal of Pure and Applied Mathematics*, 2018, vol. 119, iss. 17, pp. 1307–1319.
9. **Sharma N., Jain V., Mishra A.** An analysis of convolutional neural networks for image classification. *Procedia Computer Science*, 2018, vol. 132, pp. 377–384.
10. **Wang M., Deng W.** Deep visual domain adaptation: A survey. *Neurocomputing*, 2018, vol. 312, pp. 135–153. doi:10.1016/j.neucom.2018.05.083
11. **Nafaa S., Essam H., Ashqar H. I., Ashour K., Emad D., Hassan A. A., Mohamed R., Elhenawy M., Alhadidi T. I.** Advancing roadway sign detection with YOLO models and transfer learning. *2024 IEEE 3rd*

International Conference on Computing and Machine Intelligence (ICMI), 2024, pp. 1–4. doi:10.1109/ICMI60790.2024.10586105

12. **Linhui W., Wangpeng S., Yonghong T., Zhizhuang L., Xiongkui H., Hongyan X., Yu Y.** Transfer learning-based lightweight SSD model for detection of pests in citrus. *Agronomy*, 2023, vol. 13, iss. 7, Art. 1710. doi:10.3390/agronomy13071710
13. **Mahendrakar T., Ekblad A., Fischer N., White R. T., Wilde M., Kish B., Silver I.** Performance study of YOLOv5 and faster R-CNN for autonomous navigation around non-cooperative targets. *2022 IEEE Aerospace Conference (AERO)*, 2022, pp. 1–12. doi:10.1109/AERO53065.2022.9843537
14. **Zheng P., Zhu X., Guo W.** Brain tumour segmentation based on an improved U-Net. *BMC Medical Imaging*, 2022, vol. 22, Art. 199. doi:10.1186/s12880-022-00931-1
15. **Siddique N., Sidike P., Elkin C., Devabhaktuni V.** *U-Net and its variants for medical image segmentation: theory and applications*. https://www.researchgate.net/publication/345261676_U-Net_and_its_variants_for_medical_image_segmentation_theory_and_applications (дата обращения: 10.02.2025).
16. **You K., Liu Y., Wang J., Long M.** LogME: Practical assessment of pre-trained models for transfer learning. *Proceedings of the 38th International Conference on Machine Learning (ICML 2021). Proceedings of Machine Learning Research*, 2021, vol. 139, pp. 12133–12143.
17. **Dawson H. L., Dubrule O., Cédric M. J.** Impact of dataset size and convolutional neural network architecture on transfer learning for carbonate rock classification. *Computers & Geosciences*, 2023, vol. 171, Art. 105284. doi:10.1016/j.cageo.2022.105284
18. **Zhang Y., Tino P., Leonardis A., Tang K.** A survey on neural network interpretability. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2021, vol. 5, iss. 5, pp. 726–742. doi:10.1109/TETCI.2021.3100641
19. **Wenzel F., Dittadi A., Gehler M., Simon-Gabriel J. S., Horn M.** Assaying out-of-distribution generalization in transfer learning. *Advances in Neural Information Processing Systems*, 2022, vol. 35, pp. 7181–7198. doi:10.5555/3600270.3600791
20. **Xu H., Zhengyan Z., Ning D., Yuxian G., Xiao L., Yuqi H., Jiezhong Q., Yuan Y., Ao Z., Liang Z., Wentao H., Minlie H., Qin J., Yanyan L., Yang L., Zhiyuan L., Zhiwu L., Xipeng Q., Ruihua S., Jie T., Ji-Rong W., Jinhui Y., Wayne X. Z., Jun Z.** Pre-trained models: Past, present and future. *AI Open*, 2021, vol. 2, pp. 225–250. doi:10.1016/j.aiopen.2021.08.002
21. **Tan M., Le Q.** EfficientNet: Rethinking model scaling for convolutional neural networks. *Computer Vision and Pattern Recognition*, 2019, vol. 97, pp. 6105–6114.
22. **Yu J., Wang Z., Vasudevan V., Yeung L., Seyedhosseini M., Wu Y.** CoCa: Contrastive Captioners are image-text foundation models. *Transactions on Machine Learning Research*, 2022. arXiv.2205.01917. https://doi.org/10.48550/arXiv.2205.01917
23. **Trofimov A. G., Bogatyreva A. A.** A method of choosing a pre-trained convolutional neural network for transfer learning in image classification problems. *Advances in Neural Computation, Machine Learning, and Cognitive Research III. NEUROINFORMATICS 2019. Studies in Computational Intelligence*, Springer, Cham, 2021, vol. 856, pp. 263–270. doi:10.1007/978-3-030-30425-6_31
24. **Brostow G. J., Fauqueur J., Cipolla R.** Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, 2009, vol. 30, no. 2, pp. 88–97. doi:10.1016/j.patrec.2008.04.005
25. **Erisen S.** SERNet-Former: Segmentation by Efficient-ResNet with attention-boosting gates and attention-fusion networks. *Computer Vision and Machine Intelligence*, 2024, pp. 1–6. https://dblp.org/db/conf/cvmi/cvmi2024.html (дата обращения: 15.02.2025). doi:10.1109/CVMI61877.2024.10782648

UDC 004.89
doi:10.31799/1684-8853-2026-1-77-88
EDN: MNREWK

Criteria for selecting a transfer learning model for image analysis tasks

T. E. Shulga[a], Dr. Sc., Phys.-Math., Professor, orcid.org/0000-0002-5521-5960, taiss@yandex.ru
A. A. Sytnik[a], Dr. Sc., Tech., Professor, orcid.org/0000-0002-1256-7253, as@sstu.ru
D. A. Solopekin[a], Post-Graduate Student, orcid.org/0009-0006-7546-0150
[a]Yuri Gagarin State Technical University of Saratov, 77, Politechnicheskaia St., 410054, Saratov, Russian Federation

Introduction: The choice of an initial model for transfer learning in image analysis tasks remains a challenging problem, despite the high efficiency of the existing approach. The absence of a systematic method for selecting a base architecture limits the applicability and performance of resulting solutions, particularly in domains with restricted computational resources and specialized data. **Purpose:** To develop and validate a structured system of criteria for selecting a source model in transfer learning tasks related to image analysis. **Results:** The study identifies five primary criteria — availability, model size, domain similarity, architecture type, and adaptability — that optimize the selection process of pre-trained models. Experimental validation on the CamVid dataset for pedestrian and vehicle segmentation confirmed the effectiveness of the proposed approach. The obtained model has achieved the following metrics: loss = 0.24, mean F1 = 0.78, mean IoU = 0.70. These results are comparable to the leading existing solutions but require significantly fewer computational resources.

Practical relevance: The proposed system of selection criteria enhances reproducibility and efficiency of the implementation process of transfer learning in applied computer vision tasks such as medical diagnosis and autonomous transport. **Discussion:** Further research may focus on formalizing the proposed system into a unified methodology and automating the model selection process in order to improve its usability and objectivity.

Keywords — transfer learning, deep learning, model selection criteria, computer vision, image analysis.

For citation: Shulga T. E., Sytnik A. A., Solopekin D. A. Criteria for selecting a transfer learning model for image analysis tasks. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2026, no. 1, pp. 77–88 (In Russian). doi:10.31799/1684-8853-2026-1-77-88, EDN: MNREWK

References

1. Akutin A. S., Goriakin M. V., Zubavlenko R. A., Pechenkin V. V., Solopekin D. A. Using artificial neural networks to perform segmentation of hip radiographs in the treatment of osteoarthritis. *Modeling, Optimization and Information Technology*, 2024, vol. 12, no. 1. Available at: https://moitvivt.ru/ru/journal/pdf?id=1486 (accessed 24 December 2024) (In Russian). doi:10.26102/2310-6018/2024.44.1.011
2. Nunez A., Misra S., Falola Y. Transfer learning for geological carbon storage forecasting using neural operator. *Advances in Water Resources*, 2025, vol. 199. doi:10.1016/j.advwatres.2025.104948
3. Thanh V. N., Thang P. C., Truong V. V., Thao H. V., Quan T. M., Minh H. N. N. Traffic sign recognition through the use of an Internet of Things system and deep learning. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 2, pp. 2–15. doi:10.31799/1684-8853-2025-2-2-15, EDN: MLUUSQ
4. Shulga T. E., Solopekin D. A. Recognition of Russian-style road signs using neural networks. *Vestnik of Astrakhan State Technical University. Series: Management, Computer Science and Informatics*, 2024, no. 2, pp. 85–94 (In Russian). doi:10.24143/2072-9502-2024-2-85-94
5. Aaryan P., Damodar P., Shaswata M., Sudip M., Shahram R. Transfer learning applied to computer vision problems: survey on current progress, limitations, and opportunities. *Computer Vision and Pattern Recognition*, 2024, vol. 1, iss. 1, 16 p.
6. Xue Y., Yang R., Chen X., Liu W., Wang Z., Liu X. A review on transferability estimation in deep transfer learning. *IEEE Transactions on Artificial Intelligence*, 2024, pp. 1–24. doi:10.1109/TAI.2024.3445892
7. Zhuang F., Qi Z., Duan K., Xi D., Zhu Y., Zhu H., Xiong H., He Q. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 2021, vol. 109, no. 1, pp. 43–76. doi:10.1109/JPROC.2020.3004555
8. Ramprasath M., Anand M. V., Hariharan S. Image classification using convolutional neural networks. *International Journal of Pure and Applied Mathematics*, 2018, vol. 119, iss. 17, pp. 1307–1319.
9. Sharma N., Jain V., Mishra A. An analysis of convolutional neural networks for image classification. *Procedia Computer Science*, 2018, vol. 132, pp. 377–384.
10. Wang M., Deng W. Deep visual domain adaptation: A survey. *Neurocomputing*, 2018, vol. 312, pp. 135–153. doi:10.1016/j.neucom.2018.05.083
11. Nafaa S., Essam H., Ashqar H. I., Ashour K., Emad D., Hassan A. A., Mohamed R., Elhenawy M., Alhadidi T. I. Advancing roadway sign detection with YOLO models and transfer learning. *2024 IEEE 3rd International Conference on Computing and Machine Intelligence (ICMI)*, 2024, pp. 1–4. doi:10.1109/ICMI60790.2024.10586105
12. Linhui W., Wangpeng S., Yonghong T., Zhizhuang L., Xiongkui H., Hongyan X., Yu Y. Transfer learning-based lightweight SSD model for detection of pests in citrus. *Agronomy*, 2023, vol. 13, iss. 7, Art. 1710. doi:10.3390/agronomy13071710
13. Mahendrakar T., Ekblad A., Fischer N., White R. T., Wilde M., Kish B., Silver I. Performance study of YOLOv5 and faster R-CNN for autonomous navigation around non-cooperative targets. *2022 IEEE Aerospace Conference (AERO)*, 2022, pp. 1–12. doi:10.1109/AERO53065.2022.9843537
14. Zheng P., Zhu X., Guo W. Brain tumour segmentation based on an improved U-Net. *BMC Medical Imaging*, 2022, vol. 22, Art. 199. doi:10.1186/s12880-022-00931-1
15. Siddique N., Sidike P., Elkin C., Devabhaktuni V. *U-Net and its variants for medical image segmentation: theory and applications*. https://www.researchgate.net/publication/345261676_U-Net_and_its_variants_for_medical_image_segmentation_theory_and_applications (accessed 10 February 2025).
16. You K., Liu Y., Wang J., Long M. LogME: Practical assessment of pre-trained models for transfer learning. *Proceedings of the 38th International Conference on Machine Learning (ICML 2021). Proceedings of Machine Learning Research*, 2021, vol. 139, pp. 12133–12143.
17. Dawson H. L., Dubrule O., Cédric M. J. Impact of dataset size and convolutional neural network architecture on transfer learning for carbonate rock classification. *Computers & Geosciences*, 2023, vol. 171, Art. 105284. doi:10.1016/j.cageo.2022.105284
18. Zhang Y., Tino P., Leonardis A., Tang K. A survey on neural network interpretability. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2021, vol. 5, iss. 5, pp. 726–742. doi:10.1109/TETCI.2021.3100641
19. Wenzel F., Dittadi A., Gehler M., Simon-Gabriel J. S., Horn M. Assaying out-of-distribution generalization in transfer learning. *Advances in Neural Information Processing Systems*, 2022, vol. 35, pp. 7181–7198. doi:10.5555/3600270.3600791
20. Xu H., Zhengyan Z., Ning D., Yuxian G., Xiao L., Yuqi H., Jiezhong Q., Yuan Y., Ao Z., Liang Z., Wentao H., Minlie H., Qin J., Yanyan L., Yang L., Zhiyuan L., Zhiwu L., Xipeng Q., Ruihua S., Jie T., Ji-Rong W., Jinhui Y., Wayne X. Z., Jun Z. Pre-trained models: Past, present and future. *AI Open*, 2021, vol. 2, pp. 225–250. doi:10.1016/j.aiopen.2021.08.002
21. Tan M., Le Q. EfficientNet: Rethinking model scaling for convolutional neural networks. *Computer Vision and Pattern Recognition*, 2019, vol. 97, pp. 6105–6114.
22. Yu J., Wang Z., Vasudevan V., Yeung L., Seyedhosseini M., Wu Y. CoCa: Contrastive Captioners are image-text foundation models. *Transactions on Machine Learning Research*, 2022. arXiv.2205.01917. https://doi.org/10.48550/arXiv.2205.01917
23. Trofimov A. G., Bogatyreva A. A. A method of choosing a pre-trained convolutional neural network for transfer learning in image classification problems. *Advances in Neural Computation, Machine Learning, and Cognitive Research III. NEUROINFORMATICS 2019. Studies in Computational Intelligence*, Springer, Cham, 2021, vol. 856, pp. 263–270. doi:10.1007/978-3-030-30425-6_31
24. Brostow G. J., Fauqueur J., Cipolla R. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, 2009, vol. 30, no. 2, pp. 88–97. doi:10.1016/j.patrec.2008.04.005
25. Erisen S. SERNet-Former: Segmentation by Efficient-ResNet with attention-boosting gates and attention-fusion networks. *Computer Vision and Machine Intelligence*, 2024, pp. 1–6. doi:10.1109/CVMI61877.2024.10782648. Available at: https://dblp.org/db/conf/cvmi/cvmi2024.html (accessed 15 February 2025).

ДОЛГУШИН Михаил Дмитриевич

Аспирант, младший научный сотрудник лаборатории речевых и многомодальных интерфейсов Санкт-Петербургского Федерального исследовательского центра РАН.
В 2024 году окончил Санкт-Петербургский государственный университет по специальности «Лингвистика».
Является автором более десяти научных публикаций.
Область научных интересов — автоматическое распознавание речи, нейронные сети, объяснимый искусственный интеллект, компьютерная лингвистика.
Эл. адрес: dolgushin.m@iias.spb.su

ЕГОРОВ Елизар Валерьевич

Младший научный сотрудник Военно-космической академии им. А. Ф. Можайского, Санкт-Петербург.
В 2022 году окончил Санкт-Петербургский государственный университет аэрокосмического приборостроения по специальности «Применение и эксплуатация автоматизированных систем специального назначения».
Является автором семи научных публикаций.
Область научных интересов — информационная безопасность, обнаружение кибератак, противодействие вредоносному программному обеспечению, вирусный анализ, графовые модели.
Эл. адрес: elizarspb@yandex.ru

ЗДОРНИКОВ Егор Олегович

Аспирант, программист факультета безопасности информационных технологий Университета ИТМО, Санкт-Петербург.
В 2022 году окончил Университет ИТМО по специальности «Функциональная безопасность беспилотных транспортных средств».
Является автором пяти научных публикаций.
Область научных интересов — защита контейнеризированной и облачной инфраструктуры.
Эл. адрес: e.zdornickow2012@yandex.ru

ДРАЧЕВ Григорий Александрович

Преподаватель Департамента прикладной математики Московского института электроники и математики ВШЭ, руководитель проектов Центра специальной системотехники — сервис, Москва.
В 2018 году окончил Московский институт электроники и математики ВШЭ по специальности «Компьютерная безопасность».
Является автором шести научных публикаций.
Область научных интересов — компьютерная безопасность, методы проектирования программных систем контроля и управления информационной безопасностью, построение искусственных нейронных сетей.
Эл. адрес: pendal2@gmail.com

ЗАДБОЕВ Вадим Александрович

Младший научный сотрудник Военной академии связи им. Маршала Советского Союза С. М. Буденного, Санкт-Петербург.
В 2017 году окончил Северо-Кавказский федеральный университет по специальности «Информационная безопасность автоматизированных систем».
Является автором 34 научных публикаций и двух патентов на изобретения.
Область научных интересов — информационная безопасность, системы контроля и управления безопасностью информационно-телекоммуникационных и информационно-вычислительных систем.
Эл. адрес: zadboev89@mail.ru

КАГИРОВ Ильдар Амирович

Научный сотрудник лаборатории речевых и многомодальных интерфейсов Санкт-Петербургского Федерального исследовательского центра РАН.
В 2008 году окончил филологический факультет Санкт-Петербургского государственного университета по специальности «Лингвистика».
Является автором более 40 научных публикаций.
Область научных интересов — малоресурсные языки, человеко-машинное взаимодействие, синтаксис и грамматическая семантика естественных языков, корпусная лингвистика.
Эл. адрес: kagirov@iias.spb.su

КИПЯТКОВА Ирина Сергеевна

Доцент, старший научный сотрудник лаборатории речевых и многомодальных интерфейсов Санкт-Петербургского Федерального исследовательского центра РАН.
В 2008 году окончила Санкт-Петербургский государственный университет аэрокосмического приборостроения по специальности «Роботы и робототехнические системы».
В 2011 году защитила диссертацию на соискание ученой степени кандидата технических наук.
Является автором более 100 научных публикаций.
Область научных интересов — автоматическое распознавание речи, нейронные сети, малоресурсное распознавание языков.
Эл. адрес: kipyatkova@iias.spb.su

ЛИПАТНИКОВ Валерий Алексеевич

Профессор, старший научный сотрудник Военной академии связи им. Маршала Советского Союза С. М. Буденного, Санкт-Петербург, заслуженный изобретатель РФ, член-корреспондент РАЕН.
В 1974 году окончил Военную академию связи им. Маршала Советского Союза С. М. Буденного по специальности «Специальная радиотехника».
В 2000 году защитил диссертацию на соискание ученой степени доктора технических наук.
Является автором более 300 научных публикаций и 80 патентов на изобретения.
Область научных интересов — теория многоуровневой иерархической радиоэлектронной защиты, безопасности связи и информации инфотелекоммуникационных сетей.
Эл. адрес: lipatnikovanl@mail.ru

СЕРГЕЕВ Александр Михайлович

Доцент кафедры вычислительных систем и сетей Санкт-Петербургского государственного университета аэрокосмического приборостроения.
В 2004 году окончил Санкт-Петербургский государственный университет аэрокосмического приборостроения по специальности «Вычислительные машины, комплексы, системы и сети».
В 2020 году защитил диссертацию на соискание ученой степени кандидата технических наук.
Является автором более 100 научных публикаций, двух монографий и 11 учебных пособий.
Область научных интересов — ортогональные матрицы, ортогональные преобразования аудио- и видеоданных.
Эл. адрес: aleks.asklab@gmail.com

КУЗИН Павел Игоревич

Доцент кафедры информационных систем и технологий Санкт-Петербургского государственного лесотехнического университета им. С. М. Кирова.
В 2000 году окончил Томское высшее военное ордена Красной Звезды командное училище связи по специальности «Радиосвязь, радиовещание и телевидение», в 2008 году — Санкт-Петербургский государственный экономический университет по специальности «Финансовый менеджмент».
В 2020 году защитил диссертацию на соискание ученой степени кандидата технических наук.
Является автором 100 научных публикаций.
Область научных интересов — информационная безопасность, контроль безопасности связи, электронная техника и др.
Эл. адрес: kuzik78@mail.ru

ПОПОВ Илья Юрьевич

Доцент, директор лаборатории валидации и верификации сложных технических систем, Университет ИТМО, Санкт-Петербург.
В 2015 году окончил Университет ИТМО по специальности «Комплексная защита объектов информатизации».
В 2020 году защитил диссертацию на соискание ученой степени кандидата технических наук.
Является автором 75 научных публикаций.
Область научных интересов — fpv-системы, аппаратные решения защиты информации, безопасность сетей, видеоаналитика, игротехнический менеджмент, инструментальные средства защиты информации, комплексная защита информации, машинное обучение, перехват информации и др.
Эл. адрес: ilyapopov27@gmail.com

СОЛОДУХА Роман Александрович

Доцент кафедры информационных технологий, моделирования и управления Воронежского государственного университета инженерных технологий.
В 1998 году окончил Воронежскую высшую школу МВД России по специальности «Радиотехника».
В 2001 году защитил диссертацию на соискание ученой степени кандидата технических наук.
Является автором более 100 научных публикаций, 27 учебно-методических работ, десяти патентов на программы для ЭВМ.
Область научных интересов — стеганоанализ.
Эл. адрес: standartal@list.ru

СОЛОПЕКИН Дмитрий Андреевич

Аспирант кафедры информационно-коммуникационных систем и программной инженерии Саратовского государственного технического университета им. Гагарина Ю. А.
В 2023 году окончил магистратуру Саратовского государственного технического университета им. Гагарина Ю. А. по специальности «Программная инженерия».
Является автором 13 научных публикаций.
Область научных интересов – обработка изображений, трансферные модели искусственного интеллекта, базы данных, большие данные.
Эл. адрес:
solopekindmitriii@gmail.com

ШЕВЧЕНКО Александр Александрович

Доцент кафедры защищенных систем связи Санкт-Петербургского государственного университета телекоммуникаций им. проф. М. А. Бонч-Бруевича.
В 2015 году окончил Рязанский государственный радиотехнический университет по специальности «Компьютерная безопасность».
В 2021 году защитил диссертацию на соискание ученой степени кандидата технических наук.
Является автором 97 научных публикаций и девяти патентов на изобретения.
Область научных интересов – компьютерная безопасность, информационная безопасность, способы контроля уязвимостей и управления безопасностью информационных сетей.
Эл. адрес:
alex_pavel1991@mail.ru

СЫТНИК Александр Александрович

Профессор, заведующий кафедрой информационно-коммуникационных систем и программной инженерии Саратовского государственного технического университета им. Гагарина Ю. А. Лауреат премии Президента РФ, заслуженный деятель науки РФ.
В 1979 году окончил Саратовский государственный университет им. Н. Г. Чернышевского по специальности «Прикладная математика».
В 1993 году защитил диссертацию на соискание ученой степени доктора технических наук.
Является автором более 270 научных публикаций, в том числе 12 монографий.
Область научных интересов – дискретная математика, математические методы и модели сложных систем, современные образовательные технологии.
Эл. адрес: as@sstu.ru

ШУЛЬГА Татьяна Эриковна

Профессор кафедры информационно-коммуникационных систем и программной инженерии Саратовского государственного технического университета им. Гагарина Ю. А.
В 1997 году окончила Саратовский государственный университет им. Н. Г. Чернышевского по специальности «Прикладная математика».
В 2010 году защитила диссертацию на соискание ученой степени доктора физико-математических наук.
Является автором более 150 научных публикаций, в том числе двух монографий.
Область научных интересов – онтологический инжиниринг знаний, разработка математических моделей и методов организации управления дискретными системами и др.
Эл. адрес: taiss@yandex.ru

Уважаемые авторы!

При подготовке рукописей статей необходимо руководствоваться следующими рекомендациями.

Статьи должны содержать изложение новых научных результатов. Название статьи должно быть кратким, но информативным. В названии недопустимо использование сокращений, кроме самых общепринятых (РАН, РФ, САПР и т. п.).

Текст рукописи должен быть оригинальным, а цитирование и самоцитирование корректно оформлено.

Объем статьи (текст, таблицы, иллюстрации и библиография) не должен превышать эквивалента в 20 страниц, напечатанных на бумаге формата А4 на одной стороне через 1,5 интервала Word шрифтом Times New Roman размером 13, поля не менее двух сантиметров.

Обязательными элементами оформления статьи являются: индекс УДК, заглавие, инициалы и фамилия автора (авторов), ученая степень, звание (при отсутствии — должность), полное название организации, аннотация и ключевые слова на русском и английском языках, ORCID и электронный адрес одного из авторов. При написании аннотации не используйте аббревиатур и не делайте ссылок на источники в списке литературы. Предоставляйте подрисуночные подписи и названия таблиц на русском и английском языках.

Статьи авторов, не имеющих ученой степени, рекомендуется публиковать в соавторстве с научным руководителем, наличие подписи научного руководителя на рукописи обязательно; в случае самостоятельной публикации обязательно предоставляйте заверенную по месту работы рекомендацию научного руководителя с указанием его фамилии, имени, отчества, места работы, должности, ученого звания, ученой степени.

Простые **формулы** набирайте в Word, сложные с помощью редактора Mathtype или Equation. Для набора одной формулы не используйте два редактора; при наборе формул в формульном редакторе знаки препинания, ограничивающие формулу, набирайте вместе с формулой; для установки размера шрифта в Mathtype никогда не пользуйтесь вкладкой Other, Smaller, Larger, используйте заводские установки редактора, не подгоняйте размер символов в формулах под размер шрифта в тексте статьи, не растягивайте и не сжимайте мышью формулы, вставленные в текст; пробелы в формуле ставьте только после запятых при перечислении с помощью Ctrl+Shift+Space (пробел); не отделяйте пробелами знаки: + = – ×, а также пространство внутри скобок; для выделения греческих символов в Mathtype полужирным начертанием используйте Style → Other → bold.

Для набора формул в Word никогда не используйте вкладки: «Уравнение», «Конструктор», «Формула» (на верхней панели: «Вставка» — «Уравнение»), так как этот ресурс предназначен только для внутреннего использования в Word и не поддерживается программами, предназначенными для изготовления оригинал-макета журнала.

При наборе символов в тексте помните, что символы, обозначаемые латинскими буквами, набираются светлым курсивом, русскими и греческими — светлым прямым, векторы и матрицы — прямым полужирным шрифтом.

Подробнее см. http://i-us.ru/index.php/ius/author-guide

Иллюстрации:

— рисунки, графики, диаграммы, блок-схемы предоставляйте в виде отдельных исходных файлов, поддающихся редактированию, используя векторные программы: Visio (*.vsd, *.vsdx); Adobe Illustrator (*.ai); Coreldraw (*.cdr, версия не выше 15); Excel (*.xls); Word (*.docx); AutoCad, Matlab (экспорт в PDF, EPS, SVG, WMF, EMF); Компас (экспорт в PDF); веб-портал DRAW.IO (экспорт в PDF); Inkscape (экспорт в PDF);

— фото и растровые — в формате *.tif, *.png с максимальным разрешением (не менее 300 pixels/inch).

Наличие подрисуночных подписей и названий таблиц на русском и английском языках обязательно (желательно не повторяющих дословно комментарии к рисункам в тексте статьи).

В редакцию предоставляются:

— сведения об авторе (фамилия, имя, отчество, место работы, должность, ученое звание, учебное заведение и год его окончания, ученая степень и год защиты диссертации, область научных интересов, количество научных публикаций, домашний и служебный адреса и телефоны, e-mail), фото авторов: анфас, в темной одежде на белом фоне, должны быть видны плечи и грудь, высокая степень четкости изображения без теней и отблесков на лице, фото можно представить в электронном виде в формате *.tif, *.png, *.jpg с максимальным разрешением — не менее 300 pixels/inch при минимальном размере фото 40×55 мм;

— экспертное заключение;

— экспортное заключение.

Список литературы составляется по порядку ссылок в тексте и оформляется следующим образом:

— для книг и сборников — фамилия и инициалы авторов, полное название книги (сборника), город, издательство, год, общее количество страниц, doi;

— для журнальных статей — фамилия и инициалы авторов, полное название статьи, название журнала, год издания, номер журнала, номера страниц, doi;

— ссылки на иностранную литературу следует давать на языке оригинала без сокращений;

— при использовании web-материалов указывайте адрес сайта и дату обращения.

Список литературы оформляйте двумя отдельными блоками по образцам lit.dot на сайте журнала (http://i-us.ru/paperrules): Литература и References.

Более подробно правила подготовки текста с образцами изложены на нашем сайте в разделе «Руководство для авторов» – http://i-us.ru/index.php/ius/author-guide.

Контакты

Куда: 190000, г. Санкт-Петербург, ул. Большая Морская, д. 67, лит. А, ГУАП, РИЦ
Кому: Редакция журнала «Информационно-управляющие системы»
Тел.: (812) 494-70-02
Эл. почта: ius.spb@gmail.com
Сайт: www.i-us.ru