

## Раздел III. Алгоритмы обработки информации

УДК 004.896

DOI 10.18522/2311-3103-2022-2-165-179

Э.В. Кулиев, В.А. Семенов, А.В. Котельва, С.В. Игнатъева

### АНАЛИТИЧЕСКИЙ ОБЗОР АЛГОРИТМА ДЕРЕВА РЕШЕНИЙ В ТЕХНОЛОГИИ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ\*

*Алгоритм дерева решений является распространенным алгоритмом классификации в технологии интеллектуального анализа данных, и его результаты обычно выражаются в виде правил «если-то». Алгоритм C4.5 является одним из алгоритмов дерева решений, который обладает преимуществами простоты понимания и высокой точности, а также усовершенствован добавлением скорости прироста информации по сравнению с его предшественником - алгоритмом ID3. После теоретического анализа информации выбирается алгоритм C4.5 для анализа результатов служебной аттестации, и создается дерево решений для служебной аттестации путем сбора данных, предварительной обработки данных, расчета коэффициента прироста информации и определения атрибутов разделения. Система разработана в архитектуре B/S, а платформа управления проектом R&D, который может выполнять анализ оценки эффективности с помощью инструментов визуализации алгоритма дерева решений и динамических веб-страниц. Система включает в себя хранение информации, управление задачами, формирование отчетов, контроль полномочий и ролей, визуализацию информации и другие функциональные модули управленческой информационной системы. Они могут реализовать функции управления проектом, такие как создание и управление проектом, поток задач, заполнение и управление информацией о сотрудниках, создание системы оценки эффективности, создание отчетов различных размеров, построение управления. Используя алгоритм дерева решений в качестве основной технологии, система получает научную надежную информацию об управлении проектами с высокой точностью и реализует визуализацию данных, что может помочь предприятиям создать хорошую систему управления в эпоху больших данных. Рассмотрены управление задачами, формирование отчетов, контроль полномочий ролей, визуализация информации и другие функциональные модули управленческой информационной системы*

*Алгоритм дерева решений; интеллектуальный анализ данных; классификация; большие данные.*

E.V. Kuliev, V.A. Semenov, A.V. Kotelva, S.V. Ignateva

### ANALYTICAL REVIEW OF THE DECISION TREE ALGORITHM IN DATA INTELLIGENCE TECHNOLOGY

*The decision algorithm is the preferred filtering algorithm in data mining technology, and its results are usually chosen in the form of "if-then" rules. Algorithm C4.5 is one of the decision algorithms that takes advantage of the ease of understanding and increasing importance, and also takes advantage of the advanced information rate gain of its advanced ID3 algorithm. After the theoretical analysis of the information, the algorithm C4.5 is selected to analyze the results of performance appraisal, and enterprise performance appraisal decisions by collecting data, pre-processing data, calculating information gain and determining selection parameters. The system is*

---

\* Исследование выполнено за счет гранта Российского научного фонда № 22–21–00316, <https://rscf.ru/project/22-21-00316/> в Южном федеральном университете.

*developed in B/S architecture, an R&D project management platform that can perform evaluation analysis with decision analysis results evaluation tools and web coverage. The system includes information storage, task management, reporting, receipt and presentation control, information visualization and other functions of the management information system functions. They can realize project management functions, such as creating and managing a project, flow tasks, filling and managing information about functions, creating a performance evaluation system, creating reports of various sizes, building management. decision decision algorithm as the core technology, the system acquires scientific significant project management information with high data accuracy, and realizes visualization, which can help the enterprise to have a good management system in large areas. Task management, reporting, audit control, information visualization and other functions of the system's management reporting management functions are included.*

*Decision tree algorithm; data mining; classification; big data.*

**Введение.** С непрерывным развитием и прогрессом компьютерных технологий и технологий больших данных появляется множество новых идей и методов для управления интернет-предприятиями. Использование интернет-технологий и анализа больших данных для эффективного управления делами предприятия повышает основную конкурентоспособность предприятий, способствует прогрессу в выполнении корпоративных проектов, повышает эффективность и мотивацию сотрудников, снижает стоимость эксплуатации и обслуживания предприятий. Применение компьютеров и технологий больших данных для управления предприятием может полностью повысить эффективность управления предприятием [1]. Все больше и больше интернет-предприятий используют различные системы для управления предприятием. Эти системы могут помочь предприятиям управлять проектами и задачами, назначать работу сотрудникам. С помощью этих систем можно управлять проектами быстрее и эффективнее. В управлении бизнесом ключевую роль в развитии компании играют оценка эффективности сотрудников и анализ стоимости проектов. Справедливая система оценки может повысить мотивацию сотрудников, повысить качество выполнения проектов, повысить креативность и ответственность сотрудников. Использование технологии интеллектуального анализа данных в сочетании с системой управления корпоративными проектами на основе информации, сообщаемой сотрудниками в системе для анализа и оценки эффективности сотрудников, может помочь сотрудникам планировать и давать справочные рекомендации руководству.

В то же время, используя данные управления проектами и результаты оценки эффективности, можно проанализировать стоимость предприятия и сделать более научное распределение в соответствии со способностью сотрудников способствовать научному и устойчивому развитию предприятия. Система управления проектами R&D (Research and Development) предназначена для управления проектами интернет-предприятий, занимающихся разработкой программного обеспечения. С помощью этой системы менеджеры проектов планируют и управляют задачами, связанными с проектами разработки программного обеспечения, организуют и координируют жизненный цикл проекта и обеспечивают эффективное достижение конечной цели проекта с помощью эффективной системы управления. В статье мы используем алгоритм дерева решений в интеллектуальном анализе данных для анализа результатов оценки эффективности, полученных сотрудниками в системе, создания дерева решений для анализа результатов, которые могут помочь сотрудникам понять цель оценки эффективности, сделать дальнейшее планирование для собственной работы и повышения эффективности работы. В то же время дерево решений может непосредственно генерировать результаты служебной аттестации сотрудников, что может в будущем снизить нагрузку на руководителей предприятия, а также сделать служебную аттестацию более справедливой и прозрачной, мобилизовать мотивацию работников предприятия. По результатам аттестации

сотрудников поощряют, наказывают и ранжируют, а также определяют их вознаграждение и совмещают с расходами в управлении проектами, формируются отчеты по каждому измерению предприятия и отображаются в наглядной форме, что делает затраты предприятия яснее и помогает руководству предприятия понять положение предприятия, чтобы сформулировать следующую стратегию и разумно распределить работу сотрудников в соответствии с их способностями, тем самым повысить конкурентоспособность на рынке предприятия [2–3]. Алгоритм дерева решений, как распространенный алгоритм интеллектуального анализа данных, также широко используется во многих областях.

Исследования алгоритмов дерева решений продолжаются с целью повышения точности алгоритмов дерева решений и их объединения с соответствующими методами в других областях для получения дополнительных преимуществ. Алгоритмы дерева решений широко используются в образовании, служебной аттестации, исследованиях и других областях. В статье мы используем облегченную среду MVVM Vue.js в сочетании с языком C#, базой данных SQL Server, языком ассемблера JavaScript и другими технологиями разработки для разработки и проектирования системы управления проектами R&D. На основе алгоритма дерева решений C4.5 набор данных, созданный в системе, обучается и анализируется для создания дерева решений, связанного с оценкой производительности. Путем тестирования и анализа дерева решений, выясняем моменты, на которые руководитель проекта обращает внимание при подсчете аттестации и формируем полную систему аттестации на основе атрибутов аттестации в системе и дерева решений. Инструменты визуализации используются для создания отчетов о затратах, связанных с предприятием. Посредством управления проектами, задачами и сотрудниками система реализует функции управления проектами, такие как создание и управление проектами, поток задач, заполнение информации о сотрудниках и управление ими, создание системы служебной аттестации, создание отчетов по каждому измерению и построение кабины управления. В этом документе алгоритм дерева решений C4.5 используется в качестве базовой технологии для получения высокоточных результатов оценки производительности и эффективной системы управления проектами, визуализации данных о затратах на основе системных данных, и помочь предприятиям создать хорошую систему управления и систему оценки эффективности. С помощью этих систем можно управлять проектами быстрее и эффективнее. В управлении бизнесом ключевую роль в развитии компании играют оценка эффективности сотрудников и анализ стоимости проектов [4].

**Сбор данных.** Интеллектуальный анализ данных заключается в анализе фактических данных для получения скрытых данных, которые люди не могут видеть напрямую, которые являются большими, случайными, нечеткими и прерывистыми, а скрытые данные неизвестны и полезны. С развитием баз данных управление данными становится все более сложным, а количество генерируемых данных становится все больше и больше. В этом контексте была разработана технология интеллектуального анализа данных для извлечения информации из данных, которые нам нужны, но которые трудно найти. Этот метод в настоящее время широко используется в управлении производством, научных исследованиях, анализе рынка и инженерном проектировании [5–7]. Интеллектуальный анализ данных – это междисциплинарная дисциплина, включающая множество дисциплин, в основном интеграцию искусственного интеллекта, базы данных, статистики, технологии визуализации, и другие дисциплины для сбора и анализа данных для получения некоторой полезной информации, которая может помочь лицам, принимающим решения, принять правильное решение и снизить ненужные риски. Основные этапы показаны на рис. 1.



Рис. 1. Этапы интеллектуального анализа данных

Подготовка данных заключается в сборе и организации информации, которую необходимо извлечь, и на практике в сборе данных для ваших собственных целей или в создании ваших собственных данных из собранных наборов данных. Интеграция данных – это обработка собранных данных в соответствии с потребностями пользователя и пониманием характеристик поля, в основном недостающие части данных и очистка грязных данных в данных. Следующим шагом является выбор данных, т.е. выбор данных в базе данных и идентификация набора данных для анализа, чтобы сузить область обработки и улучшить качество интеллектуального анализа данных. Предварительная обработка данных заключается в очистке данных с помощью статистики, алгоритмического анализа и других методов, удалении ненужных шумовых данных, чтобы получить нужный нам достоверный и стандартизированный набор данных. для обеспечения целостности и непротиворечивости набора данных [6–10]. Интеллектуальный анализ данных сначала определяет цель, то есть тип знаний, которые необходимо обнаружить, затем выбирает подходящий алгоритм интеллектуального анализа данных в соответствии с определенной целью, использует алгоритм для корреляции или классификации набора данных, извлекает соответствующие знания и, наконец, выражает это в той или иной форме. Наконец, извлеченные знания анализируются, и полезная информация извлекается из знаний и отображается с помощью инструментов визуализации. Основными методами интеллектуального анализа данных являются классификация, оценка, прогнозирование, правила ассоциации и кластеризация, где классификация, оценка и прогнозирование представляют собой управляемый интеллектуальный анализ данных, который может построить модель, которая может описывать определенные атрибуты с помощью данных, а также правила ассоциации и кластерный анализ. неуправляемый интеллектуальный анализ данных, которые используют все атрибуты, чтобы найти определенную связь.

Классификация – термин относится к контролируемому обучению. Алгоритмы классификации требуют, чтобы классы определялись на основе переменных. Характеристики данных определяют, к какому классу принадлежит. Распознава-

ние образов – один из типов задач классификации, в которых входные данные (образцы) классифицируются по различным классам на основе их сходства с определенными классами.

В реальной жизни мы часто видим, как предсказывают будущие вещи / ценности / или иначе, основываясь на прошлых и настоящих данных [7, 11]. Прогнозирование также является разновидностью классификационной задачи. В зависимости от типа приложения, например прогнозирование наводнения, где зависимыми переменными являются уровень воды в реке, ее влажность, масштаб дождя и т.д. являются атрибутами.

Регрессия – статистический метод, который используется для определения взаимосвязи между переменными (x) и зависимыми переменными (y). Существует несколько типов регрессии, таких как линейная, логистическая и т.д. Линейная регрессия используется для непрерывных значений, а логистическая регрессия используется там, где существует возможность только двух событий, таких как пройдено / сбой, истина / ложь, да / нет и т.д.

Кластеризация аналогична классификации [2, 12], т.е. группирует данные. Кластеризация относится к неконтролируемому машинному обучению. Это процесс разделения данных на группы на основе схожих типов данных.

Обобщение – не что иное, как характеристика или обобщение. Он извлекает значимую информацию из данных. Он также дает сводку числовых переменных, таких как среднее значение, режим, медиана и т. д.

Правила ассоциации – основная задача Data Mining. Помогает находить подходящие закономерности и значимые выводы из базы данных. Правило ассоциации – модель, которая извлекает типы ассоциаций данных. Например, анализ рыночной корзины, в котором правила ассоциации применяются к базе данных, чтобы узнать, какие товары покупает вместе покупатель.

Различные методы интеллектуального анализа данных имеют свои собственные алгоритмы интеллектуального анализа данных, такие как алгоритм дерева решений для классификации, регрессионный анализ для прогнозирования и кластеризация K-средних для кластеризации, которые описаны на рис. 2 .

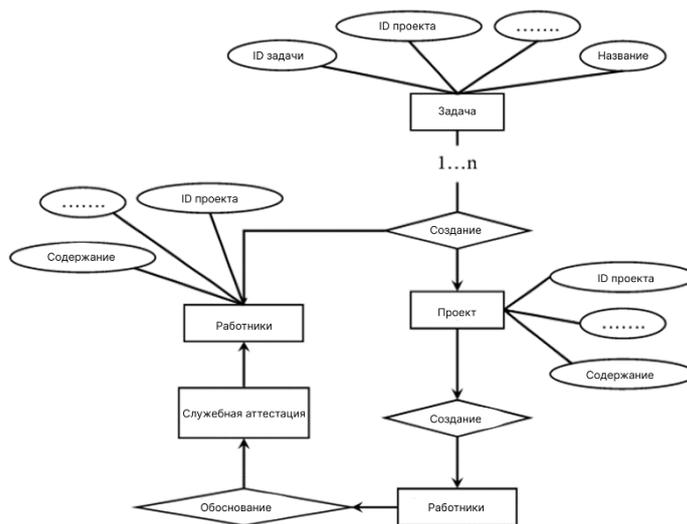


Рис. 2. Классификация алгоритмов интеллектуального анализа данных

Алгоритм, используемый в данной статье, представляет собой алгоритм дерева решений в алгоритме классификации. Среди алгоритмов классификации интеллектуального анализа данных метод классификации дерева решений имеет то преимущество, что он прост для понимания и не требует слишком много специальных базовых знаний по сравнению с другими методами классификации. Дерево решений – это алгоритм обучения с учителем, который работает как для дискретных, так и для непрерывных переменных [13]. Он разбивает набор данных на подмножества на основе наиболее значимого атрибута в наборе данных. Алгоритмы определяют, как дерево решений идентифицирует этот атрибут и как это разделение выполняется.

Наиболее значимый предиктор обозначается как корневой узел, выполняется разделение для формирования подузлов, называемых узлами решения, а узлы, которые не разделяются дальше, являются конечными или листовыми узлами.

В дереве решений набор данных разделен на однородные и неперекрывающиеся области. Он следует нисходящему подходу, поскольку верхняя область представляет все наблюдения в одном месте, которое разделяется на две или несколько ветвей, которые далее разделяются. Этот подход также называют жадным, поскольку он учитывает только текущий узел между работающими, не фокусируясь на будущих узлах. Алгоритмы дерева решений будут продолжать работать до тех пор, пока не будут достигнуты критерии остановки, такие как минимальное количество наблюдений и т.д.

После построения дерева решений многие узлы могут представлять выбросы или зашумленные данные [8]. Для удаления ненужных данных применяется метод обрезки дерева. Это, в свою очередь, повышает точность модели классификации. Для определения точности модели используется тестовый набор, состоящий из тестовых кортежей и меток классов. Процент кортежей тестового набора правильно классифицируется моделью для определения точности модели. Если модель оказывается точной, она используется для классификации кортежей данных, для которых неизвестны метки классов [14].

Результаты, генерируемые деревьями решений, обычно выражаются в форме правил «если-то», которые ясны и просты и широко используются во многих областях, таких как финансовая отрасль, метеорологический анализ и управление дорожным движением. Корневой узел представляет атрибут дерева, листья представляют маркеры классификации, а ветви представляют выходные результаты. Метод начинается с корневого узла и проходит по дереву, назначая экземпляры его дочерним элементам на основе результатов теста. Каждый дочерний узел принимает значение для этой функции и продолжает тестировать и назначать экземпляры с помощью рекурсивного метода, пока не достигнет конечного узла, где экземпляры окончательно назначаются классу конечного узла.

В дереве решений есть два типа наборов данных: набор выборочных данных и набор тестовых данных. Набор выборочных данных представляет собой набор данных, в которых известны атрибуты и классификации, и алгоритм используется для обучения набора выборочных данных для создания соответствующего дерева решений. Набор тестовых данных используется для проверки сгенерированного дерева решений, внесения данных в дерево решений, получения окончательных категорий, сравнения с фактическими типами и измерения точности дерева решений.

Алгоритм дерева решений эффективен, прост для понимания, имеет небольшой объем вычислений и хорошо справляется с дискретными данными. О дереве решений можно судить на основе его правильности, его эффективности после тестирования выборочного набора данных, а также его сложности, косвенности и

масштаба [15]. Информационная энтропия является ключевым элементом алгоритмов дерева решений, где слово «энтропия» используется в термодинамике и представляет собой меру степени хаоса в системе в физике.

В 1948 г. Шеннон, отец теории информации, заимствовал понятие энтропии и ввел информационную энтропию, которая определяется как вероятность возникновения дискретных случайных событий. Вообще говоря, чем выше вероятность появления сообщения, тем больше оно цитируется, и чем ниже информационная энтропия, тем выше вероятность появления сообщения. Формула расчета информационной энтропии определяется следующим образом: его эффективность после тестирования выборочного набора данных, а также его сложность, косвенность и масштаб.

$$G(x) = \lim_{n \rightarrow \infty} \sum_{i=1}^n \frac{-q_i}{\ln q_i} \quad (1)$$

$$q_i = \frac{\pi}{\sqrt{x+y}} \quad (2)$$

Алгоритмы дерева решений обычно состоят из трех шагов: выбор признаков, создание дерева решений и обрезка дерева решений. Обычно используемые алгоритмы дерева решений включают алгоритм ID3 и алгоритм C4.5, основанный на максимальной скорости получения информации, который улучшен по сравнению с алгоритмом ID3, и CART, основанный на индексе Джини.

**Оптимизация модели дерева решений.** Алгоритм основан на концепции получения информации для выбора классификационных признаков дерева решений. Величина прироста информации представляет собой разницу между уровнем примеси выборочных данных до классификации и уровнем примеси после классификации. Алгоритм ID3 основан на теории информации и использует информационную энтропию и прирост информации в качестве критериев для классификации существующих наборов данных. При построении дерева решений учитывается прирост информации для каждого выбора узла ветвления [11, 16]. Прирост информации всех атрибутов вычисляется и сравнивается, и атрибут с наибольшим приростом информации берется в качестве атрибута разделения, а последующие листовые узлы продолжают циклически выполнять эту операцию для создания дерева решений. Алгоритм ID3 может работать только с дискретными атрибутами,  $n$  категорий, а вероятность каждой категории равна  $P(C_1), P(C_2), \dots, P(C_n)$ , и тогда информационная энтропия атрибута  $A$  после разбиения будет следующей:

$$P(C_i) = \frac{\sqrt{\lim_{n \rightarrow \infty} \sum_{i=1}^n c_i^2}}{\lim_{n \rightarrow \infty} \sum_{i=1}^n c_i^2}. \quad (3)$$

$$G(x) = \lim_{n \rightarrow \infty} \sum_{i=1}^n P(C_i). \quad (4)$$

$$G(x) = \frac{\sqrt{c_1 + c_2 + \dots + c_n}}{\lim_{n \rightarrow \infty} \sum_{i=1}^n c_i^2}. \quad (5)$$

Третий шаг требует прироста информации, который представляет собой разницу между информационной энтропией, полученной на первом этапе, и информационной энтропией, полученной на втором этапе для атрибута  $C$ . Формула выглядит следующим образом:

$$G(A) = Q(x) - Q_i(x). \quad (6)$$

$$Q(x) = \lim_{n \rightarrow \infty} \sum_{i=1}^n q_i. \quad (7)$$

Чем выше прирост информации, тем более подходит атрибут для классификации, и атрибут выбирается в качестве текущего узла. Столбцы атрибута исключаются из списка, а остальные данные перебираются с первого шага. Когда в классификации имеется только одно значение целевого атрибута или доля всех значений атрибута достигает порога, итерация завершается. Окончательное дерево решений формируется после завершения итерации.

После выполнения алгоритма C4.5 получают результат скорости прироста информации всех атрибутов, и в качестве разделяемого атрибута текущего узла выбирается атрибут с наивысшей скоростью прироста информации. Остальные атрибуты будут продолжать вычисляться рекурсивно. По мере того, как атрибуты постепенно вычисляются, скорость получения информации становится все меньше и меньше, и в качестве атрибута классификации выбирается атрибут с относительно большей скоростью получения информации. Блок-схема C4.5 показана на рис. 3.

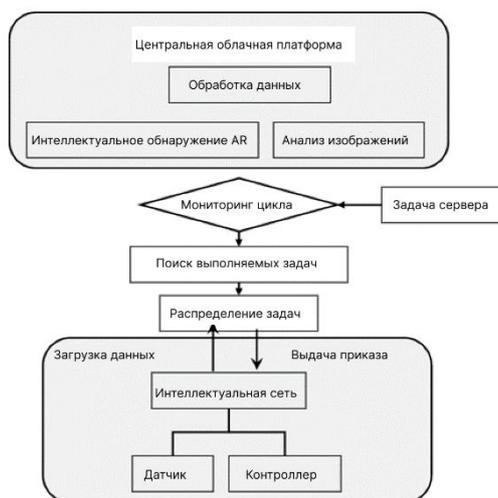


Рис. 3. Схема алгоритма C4.5

**Обрезка дерева решений.** Обучающая выборка является ключевым фактором в построении дерева решений. Когда обучающая выборка слишком мала или есть проблемы с данными, результирующее дерево решений может быть аномальным, что может привести к неточным деревьям решений и может сделать дерево решений очень сложным. Согласно исследованию, не все сложные и большие деревья решений приводят к более точным наборам правил. Следовательно, сложные деревья решений необходимо упрощать, что называется обрезкой [17]. Предварительная обрезка происходит во время построения дерева решений, когда рост прекращается в начале процесса вычислений и выполняется обрезка. Предварительная обрезка проста, но трудно определить время завершения во время роста дерева решений, что делает ее не очень практичной. Общий алгоритм дерева решений использует обрезку. Обрезка выполняется на полностью сформированном дереве решений восходящим образом, заменяя поддеревья узлов конечными узлами, не отвечающими доверительному уровню, и помечая класс наиболее часто используемым классом в поддереве узлов. Этап обрезки повторяется до тех пор, пока все узлы не будут удовлетворены условию, и итоговое сгенерированное дерево решений не станет более надежным, чем предыдущее дерево решений [18]. По сравнению с предварительной обрезкой, обрезка уменьшает количество предполагаемых вмешательств.

В настоящее время большинство существующих информационных систем управления проектами разрабатываются и проектируются с использованием архитектуры V/S, которая представляет собой модель архитектуры браузер-сервер, и пользователи обычно получают доступ к системе через браузер. Система V/S имеет преимущества простоты обслуживания и обновления, низкой стоимости, безопасности и т.д. Внешний интерфейс использует Vue.js, восходящую прогрессив-

ную среду MVVM для создания пользовательских интерфейсов, а сервер использует C#, объектно-ориентированный фреймворк, ориентированный язык разработки, производный от C++.

MVC – аббревиатура от Model-View-Controller, которая представляет собой структуру для многоуровневых систем, разделяющую бизнес-логику, данные и интерфейсы отображения. Платформа накладывается на систему, а бизнес-логика, данные и интерфейс отображения разделены, чтобы более четко разграничить их соответствующие функции. Уровень модели обычно используется для обработки логической части данных приложения, обычно отвечающей за доступ к базе данных. Слои представления – это слои отображения, который обрабатывает отображаемую часть данных и создается на основе данных модели [19]. Уровень контроллера – часть взаимодействия с пользователем, которая считывает данные со слоя представления, контролирует ввод данных пользователем и отправляет их на уровень модели.

Платформа MVC обладает такими преимуществами, как низкая связанность, возможность многократного использования, низкая стоимость жизненного цикла, высокая ремонтопригодность и быстрое развертывание. Однако это увеличивает сложность структуры и реализации системы, а для некоторых простых страниц использование фреймворка MVC может привести к снижению эффективности его работы; в то же время может быть неэффективный доступ представления к модели; тесная связь между представлением и контроллером приведет к их независимому повторному использованию. Фреймворк MVVM – это аббревиатура от Model-View-ViewModel, которая представляет собой улучшенную версию, основанную на фреймворке MVC и фреймворке MVP. Это очень затрудняет поддержку шаблона MVC. Это связано с тем, что контроллер для обработки логики и преобразования данных в сложных проектах становится очень большим и сложным в обслуживании, поэтому, чтобы изменить ограничения этой модели, логическая обработка и преобразование данных контроллера убраны из него, а для управления этими операциями создан специальный объект ViewModel. Такой подход делает код контроллера очень маленьким и простым в управлении. Это делает инфраструктуру MVVM основной платформой, используемой при разработке программного обеспечения на данном этапе.

**Анализ оптимизации на основе алгоритма дерева решений.** Справедливая и ответственная система оценки будет стимулировать мотивацию сотрудников, повышать эффективность и мотивацию сотрудников, повышать конкурентоспособность предприятия. Сегодня, с быстрым развитием Интернета, оценка эффективности с помощью интеллектуального анализа данных станет новым способом [20, 21]. Когда сотрудник отправляет задачу с помощью этой системы, система будет фиксировать время выполнения задачи и количество кодов, отправленных сотрудником, а руководитель проекта будет оценивать сотрудника на основе информации, предоставленной сотрудником. Используя эти данные, можно проанализировать производительность сотрудника с помощью алгоритма дерева решений. Перед интеллектуальным анализом данных сначала определяются выборочные данные, и данные выбираются из информации, заполняемой сотрудниками, и оценок персонала, введенных менеджерами. Поскольку компания занимается разработкой программного обеспечения, данные в основном собирались у сотрудников компании, занимающихся исследованиями и разработками. Было собрано 270 данных, 180 из которых использовались в качестве обучающего набора данных для алгоритма дерева решений, а остальные использовались в качестве тестового набора данных. Набор данных в основном состоит из таблицы информации о сотрудниках системы, рабочей ситуации сотрудника в соответствии с задачей и оценки сотрудника менеджером за период времени. Информационная таблица сотрудников: в основном это информация о сотрудниках при их регистрации в сис-

теме, среди которых критерии оценки эффективности, следующие: ФИО, отдел, специальность, степень и т.д. Статус работы: этот набор данных определяется представлением задач сотрудника, такие как заполнение часов, завершение работы, эффективность выполнения работы и т. д. Эти данные автоматически рассчитываются системой после заполнения данных сотрудником и сохраняются в разных таблицах системы и резюмируются операторами SQL. Оценка сотрудников: эти данные оцениваются руководством проекта в зависимости от результатов работы подчиненных сотрудников.

В процессе получения набора данных есть много данных с нулевым значением. Для этих данных мы оцениваем данные перед вводом значения переменной в системную базу данных и запрещаем отображение данных как нулевых [3]. Многие из приведенных выше значений переменных извлекаются из других таблиц базы данных в системе, и когда мы получим эти данные, мы заполним пустые значения перед вводом, и правила заполнения, следующие: количество кода: эти данные равны нулю, что означает у сотрудника нет записи о отправке кода в этот период времени, и он записывается как 0 значение. Данные пустые, значит сотрудник не заполнил часы за этот период времени. Поскольку уровень заполнения связан с посещаемостью сотрудника, если он пуст, сотрудник может найти свои записи о посещаемости в листе посещаемости и рассчитать процент заполнения [9]. Способность к обучению, технические способности, эффективность работы: если эти данные пусты, руководитель сотрудника не оценивал его, а оценка по умолчанию – средний балл.

В процессе сбора данных могут быть некоторые случайные ошибки или ошибки, которые называются зашумленными данными. Появление зашумленных данных может привести к большим ошибкам в результатах интеллектуального анализа данных. Следовательно, нам нужно предварительно обработать зашумленные данные; обычно методы работы с зашумленными данными, следующие: регрессия, упаковка, проверка комбинации человек-компьютер, кластеризация и т. д. Например, когда данные объема кода ненормальны, среднее значение данных может использоваться для замены данные о шуме. Кроме того, можно использовать принцип алгоритмов множественной регрессии для сглаживания шумовых данных.

Поскольку алгоритм дерева решений не может обрабатывать непрерывные значения, необходимо дискретизировать непрерывные переменные перед интеллектуальным анализом данных, как показано на рис. 4. Путем тестирования и анализа дерева решений мы выясняем моменты, на которые руководитель проекта обращает внимание при подсчете аттестации, и формируем полную систему аттестации на основе атрибутов аттестации в системе и дереве решений.

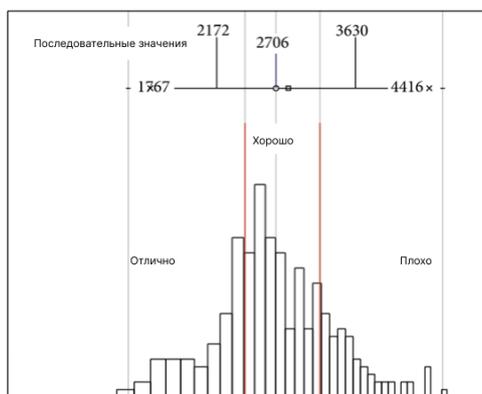


Рис. 4. Последовательные значения коэффициента заполнения работы

В данной работе берется отчет сотрудников компании, использующих эту систему в определенном месяце 2021г., и в соответствии с описанным выше процессом обработки данных мы получаем набор данных на рис.5, а данные в таблице взяты из оценочных баллов 30 сотрудников отдела разработки программного обеспечения компании в январе 2021г. В процессе собственно расчета возьмем данные 30 сотрудников этого отдела в первой половине 2021г. для алгоритма дерева решений. В процессе фактического расчета мы возьмем данные 30 сотрудников отдела в первом полугодии 2021г. для расчета алгоритма дерева решений и 90 данных в третьем квартале 2021г. для проверки результатов алгоритма дерева решений. Система может автоматически генерировать деревья решений на основе набора данных, и посредством анализа деревьев решений мы можем выяснить, какие модули беспокоят менеджеров при оценке эффективности – объем кода и завершение работы. Эти данные получены из информации, сообщаемой сотрудниками в системе, которая требует, чтобы сотрудники планировали свои задачи в своей повседневной работе, выполняли их вовремя и своевременно сообщали о своих расходах в систему. Сгенерированный алгоритм дерева решений записывается системой и впоследствии используется для оценки служебной аттестации.

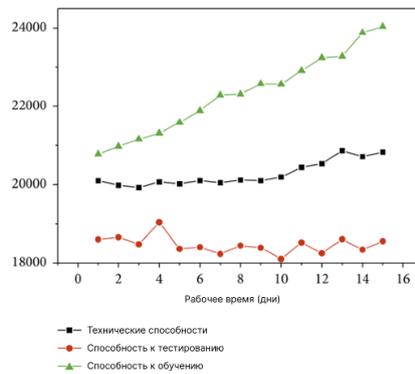


Рис. 5. Набор данных для обучения алгоритму

Целью примера является окончательный результат оценки служебной аттестации, а классом целей является «результат оценки» с четырьмя значениями: А, В, С и D. В 180 извлеченных наборах данных четыре значения следующие: 59, 92, 25 и 4.

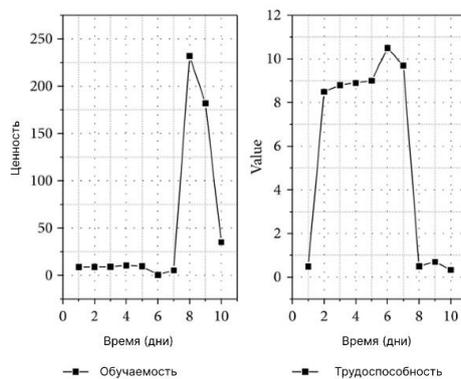


Рис. 6. Набор тестовых данных

Результаты оценки 90 тестовых образцов сравниваются с результатами оценки, полученными с помощью алгоритма дерева решений C4.5 и алгоритма дерева решений ID3, из которых 30 тестовых образцов перечислены здесь для экономии места. После создания каждого проекта участники проекта должны создать несколько задач в рамках проекта. Типы задач могут быть созданы системным администратором. В пакете задач также необходимо оценить затраты, которые будут израсходованы в рамках этого модуля. После создания задачи участники проектной группы, которым необходимо выполнить задачу, должны создать карточки задач под задачей, которые определены как краткосрочные планы и цели для участников проекта в процессе выполнения задачи. Модуль управления персоналом является ядром этой статьи. а ключевой функцией исследования является система оценки эффективности сотрудников, основанная на расчете дерева решений. В модуле управления сотрудниками, во-первых, сотрудники создают свои собственные учетные записи и заполняют информацию [2]. Затем системный администратор заполняет организацию сотрудника и другую информацию о сотруднике.

Существует два вида управления персоналом: один – это управление подчиненным персоналом руководителем проекта, а другой – самоуправление персоналом. Во-первых, сотрудники могут назначать свои задачи и устанавливать свои карточки в системе, а в системе есть персональный верстак, где сотрудники могут просматривать свои задачи и карточки и самостоятельно управлять статусом карточек задач. Под рабочим местом сотрудники могут заполнять свое рабочее время и просматривать свои еженедельные и ежемесячные отчеты. Модуль служебной аттестации является подмодулем модуля управления персоналом, который генерирует три ключевые данные: объем кода, завершение работы и скорость выполнения работы на основе информации, сообщаемой персоналом в системе. и руководитель проекта должен оценить способность к обучению и технические способности персонала на основе их обычной работы в системе. Эти индикаторы используются для создания дерева решений по оценке эффективности. Система требует, чтобы руководитель проекта дал окончательную оценку служебной аттестации сотрудника. После того, как необработанные данные доступны, система создает дерево решений, связанное с оценкой производительности, на основе набора данных, которое помогает сотруднику понять области, которые необходимо улучшить. Он будет определять заработную плату работника. система генерирует дерево решений, связанное с оценкой эффективности, на основе набора данных, которое помогает сотруднику понять области, которые необходимо улучшить. Он будет определять заработную плату работника. система генерирует дерево решений, связанное с оценкой эффективности, на основе набора данных, которое помогает сотруднику понять области, которые необходимо улучшить. Он будет определять заработную плату работника.

**Заключение.** В данной статье, с точки зрения управления проектами, алгоритм дерева решений C4.5 используется для анализа оценки производительности. Собирается и обрабатывается информация, связанная с производительностью руководителей и сотрудников проектов R&D программного обеспечения, для формирования набора данных о производительности, а затем генерируем модель дерева решений, связанную с оценкой производительности, на основе этого набора данных посредством повторной обработки данных, разделения атрибутов, построения модели, операция обрезки и оценочный анализ. Модель дерева решений была протестирована с тестовым набором данных, сгенерированным системой, уровень точности достиг более 90%, что выше, чем результат дерева решений алгоритма ID3. На основе анализа дерева решений, можно определить области, которые необходимо улучшить в работе по разработке программного обеспечения, и

предоставить ссылку для оценки эффективности менеджеров проектов. С другой стороны, информация об управлении проектами и оценка эффективности сотрудников в системе будет оказывать влияние на стоимость компании. На основе полученной информации для компании могут быть созданы отчеты о затратах. Отчеты визуализируются в многомерном виде, чтобы показать стоимость компании. Отчеты могут использоваться руководителями проектов и соответствующими руководителями компаний для четкого понимания деятельности компании, и могут быть использованы для руководства при следующем планировании компании. Статья основана на Vue.js, облегченной среде MVVM, и использует язык C#, базу данных SQL Server и другие инструменты для проектирования системы.

Система основана на C4.5 алгоритме дерева решений для сбора и анализа набора данных, сгенерированного системой, создания деревьев решений, связанных с оценкой производительности, использования Echarts и других инструментов визуализации для создания соответствующих отчетов о затратах. Система разработана в архитектуре B/S, а система управления проектами R&D и платформы, которые могут выполнять анализ оценки эффективности, построены с помощью инструментов визуализации, алгоритма дерева решений и динамических веб-страниц.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Курейчик В.В., Курейчик В.М., Родзин С.И. Теория эволюционных вычислений. – М.: Физматлит, 2012. – 260 с.
2. Штовба Д.С. Муравьиные алгоритмы: теория и применение // Математика в приложениях. – 2004. – С. 70-75.
3. Борознов В.О. Исследование решения задачи коммивояжера // Вестник Астраханского государственного технического университета. Управление, вычислительная техника и информатика. – 2009. – С. 147-151.
4. Курейчик В.М., Курейчик В.В. Генетический алгоритм разбиения графа // Известия Российской академии наук. Теория и системы управления. – 1999. – № 4. – С. 79-87.
5. Wang W., Liu L., Liu J., and Chen Z. Energy management and optimization of vehicle-to-grid systems for wind power integration // CSEE Journal of Power and Energy Systems. – 2020. – Vol. 7, No. 1. – P. 172-180.
6. Wu W., Wu W., and Wang S. Thermal management optimization of a prismatic battery with shape-stabilized phase change material // International Journal of Heat and Mass Transfer. – 2018. – Vol. 121. – P. 967-977.
7. Курейчик В.М., Лебедев Б.К., Лебедев О.К. Поисковая адаптация: теория и практика. – М.: Физматлит, 2006. – 272 с. – ISBN 5-9221-0749-6.
8. Кулиев Э.В., Лежебоков А.А., Кравченко Ю.А. Роевой алгоритм поисковой оптимизации на основе моделирования поведения летучих мышей // Известия ЮФУ. Технические науки. – 2016. – № 7 (180). – С. 53-62.
9. Курейчик В.М., Курейчик В.В., Родзин С.И. Модели параллелизма эволюционных вычислений // Вестник Ростовского государственного университета путей сообщения. – 2011. – № 3 (43). – С. 93-97.
10. Бова В.В., Курейчик В.В. Интегрированная подсистема гибридного и комбинированного поиска в задачах проектирования и управления // Известия ЮФУ. Технические науки. – 2010. – № 12 (113). – С. 37-42.
11. Кривенко М.П., Семенова М.М., Семенов В.А. Разработка принципов интеллектуального принятия решений на основе биоинспирированной оптимизации // Технологии разработки информационных систем ТРИС-2020: Матер. X Международной научно-технической конференции, Таганрог, 05–10 октября 2020 года. – Таганрог: ЮФУ, 2020. – С. 100-106.
12. Al-Falahi M.D.A., Nimma K.S., Jayasinghe S.D.G., Enshaei H., and Guerrero J.M. Power management optimization of hybrid power systems in electric ferries // Energy Conversion and Management. – 2018. – Vol. 172. – P. 50-66.

13. Bourbon R., Ngueveu S.U., Roboam X., Sareni B., Turpin C., and Hernandez-Torres D. Energy management optimization of a smart wind power plant comparing heuristic and linear programming methods // *Mathematics and Computers in Simulation*. – 2019. – Vol. 158. – P. 418-431.
14. Bray M., Wang W., Rees M.A. et al. KPDGUI: an interactive application for optimization and management of a virtual kidney paired donation program // *Computers in Biology and Medicine*. – 2019. – Vol. 108. – P. 345-353.
15. Byrne R.H., Nguyen T.A., Copp D.A., Chalamala B.R., and Gyuk I. Energy management and optimization methods for grid energy storage systems // *IEEE Access*, vol. 6, pp. 13231–13260, 2018.
16. Kursitys I., Kravchenko Y., Kuliev E., Natskevich A. A bioinspired algorithm for improving the effectiveness of knowledge processing // *Advances in Intelligent Systems and Computing* (см. в книгах). – 2021. – Vol. 1197 AISC. – P. 1491-1498.
17. Kuliev E.V., Zaporozhets D.Y., Kureichik V.V., Kursitys I.O. Wolf pack algorithm for solving vlsi design tasks // *Journal of Physics: Conference Series*. Сер. "International Conference "Information Technologies in Business and Industry" - 1 - Microprocessor Devices, Telecommunication and Networking" 2019. С. 022009.
18. Kuliev E.V., Kureichik V.V., Kursitys I.O. Decision making in VLSI components placement problem based on grey wolf optimization // 2019 IEEE East-West Design and Test Symposium, EWDTS 2019. – 2019. – P. 8884371
19. Курейчик В.В., Кулиев Э.В., Курейчик В.В. Модель адаптивного поведения "обезьян" для решения задачи компоновки блоков ЭВА // *Информатизация и связь*. – 2018. – № 4. – С. 31-37.
20. Kuliev E., Kureichik V., Kureichik V. Monkey search algorithm for ece components partitioning // *Journal of Physics: Conference Series*. International Conference Information Technologies in Business and Industry 2018 - Enterprise Information Systems. – 2018. – P. 042026.
21. Bova V.V., Kravchenko Y.A., Kureichik V.V. Development of distributed information systems: ontological approach // *Advances in Intelligent Systems and Computing*. – 2015. – Vol. 349. – P. 113-122.

## REFERENCES

1. Kureychik V.V., Kureychik V.M., Rodzin S.I. *Teoriya evolyutsionnykh vychisleniy* [Theory of evolutionary calculations]. Moscow: Fizmatlit, 2012, 260 p.
2. Shtovba D.S. Murav'inye algoritmy: teoriya i primeneniye [Ant algorithms: theory and application], *Matematika v prilozheniyakh* [Mathematics in applications], 2004, pp. 70-75.
3. Boroznov V.O. Issledovaniye resheniya zadachi kommvoyazhera [Investigation of the solution of the traveling salesman problem], *Vestnik Astrakhanskogo gosudarstvennogo tekhnicheskogo universiteta. Upravleniye, vychislitel'naya tekhnika i informatika* [Bulletin of the Astrakhan State Technical University. Management, computer engineering and computer science], 2009, pp. 147-151.
4. Kureychik V.M., Kureychik V.V. Geneticheskii algoritm razbiveniya grafa [Genetic algorithm of graph partitioning], *Izvestiya Rossiyskoy akademii nauk. Teoriya i sistemy upravleniya* [Proceedings of the Russian Academy of Sciences. Theory and control systems], 1999, No. 4, pp. 79-87.
5. Wang W., Liu L., Liu J., and Chen Z. Energy management and optimization of vehicle-to-grid systems for wind power integration, *CSEE Journal of Power and Energy Systems*, 2020, Vol. 7, No. 1, pp. 172-180.
6. Wu W., Wu W., and Wang S. Thermal management optimization of a prismatic battery with shape-stabilized phase change material, *International Journal of Heat and Mass Transfer*, 2018, Vol. 121, pp. 967-977.
7. Kureychik V.M., Lebedev B.K., Lebedev O.K. Poiskovaya adaptatsiya: teoriya i praktika [Search adaptation: theory and practice]. Moscow: Fizmatlit, 2006, 272 p. ISBN 5-9221-0749-6.
8. Kuliev E.V., Lezhebokov A.A., Kravchenko Yu.A. Royevoy algoritm poiskovoy optimizatsii na osnove modelirovaniya povedeniya letuchikh myshey [Swarm search engine optimization algorithm based on bat behavior modeling], *Izvestiya YuFU. Tekhnicheskiye nauki* [Izvestiya SFedU. Engineering Sciences], 2016, No. 7 (180), pp. 53-62.
9. Kureychik V.M., Kureychik V.V., Rodzin S.I. Modeli parallelizma evolyutsionnykh vychisleniy [Models of parallelism of evolutionary computations], *Vestnik Rostovskogo gosudarstvennogo universiteta putey soobshcheniya* [Bulletin of the Rostov State University of Railways], 2011, No. 3 (43), pp. 93-97.

10. Bova V.V., Kureychik V.V. Integrirovannaya podsystema gibridnogo i kombinirovannogo poiska v zadachakh proyektirovaniya i upravleniya // *Izvestiya YuFU. Tekhnicheskiye nauki* [Izvestiya SFedU. Engineering Sciences], 2010, No. 12 (113), pp. 37-42.
11. Krivenko M.P., Semenova M.M., Semenov V.A. Razrabotka printsiptov intellektual'nogo prinyatiya resheniy na osnove bioinspirirovannoy optimizatsii [Development of the principles of intelligent decision-making based on bioinspired optimization], *Tekhnologii razrabotki informatsionnykh sistem TRIS-2020: Mater. X Mezhdunarodnoy nauchno-tekhnicheskoy konferentsii, Taganrog, 05–10 oktyabrya 2020 goda* [Technologies for the development of information systems TRIS-2020: Materials of the X International Scientific and Technical Conference, Taganrog, October 05-10, 2020]. Taganrog: Yuzhnyy federal'nyy universitet, 2020, pp. 100-106.
12. Al-Falahi M.D.A., Nimma K.S., Jayasinghe S.D.G., Enshaei H., and Guerrero J.M. Power management optimization of hybrid power systems in electric ferries, *Energy Conversion and Management*, 2018, Vol. 172, pp. 50-66.
13. Bourbon R., Ngueveu S.U., Roboam X., Sareni B., Turpin C., and Hernandez-Torres D. Energy management optimization of a smart wind power plant comparing heuristic and linear programming methods, *Mathematics and Computers in Simulation*, 2019, Vol. 158, pp. 418-431.
14. Bray M., Wang W., Rees M.A. et al. KPDGUI: an interactive application for optimization and management of a virtual kidney paired donation program, *Computers in Biology and Medicine*, 2019, Vol. 108, pp. 345-353.
15. Byrne R.H., Nguyen T.A., Copp D.A., Chalamala B.R., and Gyuk I. Energy management and optimization methods for grid energy storage systems, *IEEE Access*, 2018, Vol. 6, pp. 13231-13260.
16. Kursitys I., Kravchenko Y., Kuliev E., Natskevich A. A bioinspired algorithm for improving the effectiveness of knowledge processing, *Advances in Intelligent Systems and Computing (sm. v knigakh)*, 2021, Vol. 1197 AISC, pp. 1491-1498.
17. Kuliev E.V., Zaporozhets D.Y., Kureichik V.V., Kursitys I.O. Wolf pack algorithm for solving vlsi design tasks, *Journal of Physics: Conference Series. Ser. "International Conference "Information Technologies in Business and Industry" - I - Microprocessor Devices, Telecommunication and Networking" 2019*, pp. 022009.
18. Kuliev E.V., Kureichik V.V., Kursitys I.O. Decision making in VLSI components placement problem based on grey wolf optimization, *2019 IEEE East-West Design and Test Symposium, EWDTS 2019*, 2019, pp. 8884371
19. Kureychik V.V., Kuliyeu E.V., Kureychik V.V. Model' adaptivnogo povedeniya "obez'yan" dlya resheniya zadachi komponovki blokov EVA [Model of adaptive behavior of "monkeys" for solving the problem of EVA block layout], *Informatizatsiya i svyaz' [Informatization and communication]*, 2018, No. 4, pp. 31-37.
20. Kuliev E., Kureichik V., Kureichik V. Monkey search algorithm for ece components partitioning, *Journal of Physics: Conference Series. International Conference Information Technologies in Business and Industry 2018 - Enterprise Information Systems*, 2018, pp. 042026.
21. Bova V.V., Kravchenko Y.A., Kureichik V.V. Development of distributed information systems: ontological approach, *Advances in Intelligent Systems and Computing*, 2015, Vol. 349, pp. 113-122.

Статью рекомендовал к опубликованию к.т.н., доцент С.Г. Буланов

**Кулиев Эльмар Валерьевич** – Южный федеральный университет; e-mail: ekuliev@sfedu.ru; г. Таганрог, Россия, тел.: 88634371651; кафедра систем автоматизированного проектирования; доцент.

**Семенов Виктор Алексеевич** – e-mail: viksemenov@sfedu.ru; тел.: 88634371651; студент.

**Котельва Ангелина Васильевна** – e-mail: kotelva@sfedu.ru; тел.: 88634371651; студент

**Игнатьева Сабина Валерьевна** - e-mail: signateva@sfedu.ru; тел.: 88634371651; студент

**Kuliev Elmar Valerievich** – Southern Federal University; e-mail: ekuliev@sfedu.ru; Taganrog, Russia; phone: +78634371651; the department of computer aided design; associate professor.

**Semenov Viktor Alekseevich** – e-mail: visemenov@sfedu.ru; phone: 78634371651; student.

**Kotelva Angelina Vasilevna** – e-mail: kotelva@sfedu.ru; phone: 88634371651; student

**Ignateva Sabina Valerievna** - e-mail: signateva@sfedu.ru; phone: 88634371651; student.