межвузовский сборник
научных трудов

# МАТЕМАТИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

2023

межвузовский сборник научных трудов

# МАТЕМАТИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

РГРТУ, 2023

**УДК 004.4**

# Редакционная коллегия



*Рецензент: к.т.н., доцент, зав. каф. «Информатики, информационных технологий и защиты информации» Липецкого государственного педагогического института Скуднев Д.М.*

**Математическое и программное обеспечение вычислительных систем**: Межвуз. сб. науч. тр. / Под ред. Г.В. Овечкина – Рязань: РГРТУ им. В.Ф. Уткина, январь 2023-124 с.

**ISBN 978-5-7722-0366-8**

Представлены статьи, посвящённые различным аспектам разработки программных средств ЭВМ, вычислительных систем и сетей, вопросам математического моделирования, методам обработки информации.

Предназначен для научно-педагогических работников вузов, инженеров, аспирантов и студентов старших курсов.

Авторская позиция и стилистические особенности публикаций полностью сохранены.

**ISBN 978-5-7722-0366-8**

# ВВЕДЕНИЕ

Межвузовский сборник научных трудов содержит результаты научных исследований и разработок по следующим направлениям:

– программное обеспечение вычислительных систем, новые информационные технологии;

– прикладная математика, теория информации, искусственный интеллект;

– ЭВМ в системах обработки, управления и обучения;

– автоматизированное проектирование аппаратных средств вычислительных систем;

– информационные технологии в экономических и социальных системах.

Сборник сформирован на основе статей, в которых рассматриваются различные аспекты разработки программных средств ЭВМ, вычислительных систем и сетей, включая вопросы автоматизации проектирования, теории обработки информации и математического моделирования, и предназначен для студентов, аспирантов и преподавателей технических вузов и научных работников.

Материалы для сборника предоставлены сотрудниками и студентами:

– Сочинского государственного университета, Сочи;

– Федерального государственного бюджетного образовательного учреждения высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань.

**УДК 004.658**

**Аксютин Р.О.**

# УЧЁТ И ПРОВЕДЕНИЕ КОРПОРАТИВНЫХ СОБРАНИЙ НА ПРЕДПРИЯТИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В статье разбирается программное обеспечение чёта и проведения корпоративных собраний на предприятии и создание на основе 1С:Предприятие системы, позволяющей их оптимизировать.

В деловых отношениях многое зависит от личных встреч, бесед, совещаний. Они, с одной стороны, помогают найти общее решение, а с другой – держать определённую дистанцию, сглаживать острые углы и с достоинством выходить из неприятных и затруднительных ситуаций, которые могут возникнуть в коллективе.

Поскольку в повседневной работе организации происходит крайне много событий, требующих внимания, совещания дают возможность заинтересованным сторонам согласовать цели, убедиться, что все находятся на одной странице, а также участвовать в стратегических обсуждениях.

Обеспечение того, чтобы все работники могли работать синхронно друг с другом, часто является сложной задачей. Также не всегда является возможным, чтобы каждый участник физически присутствовал на собрании, что добавляет ещё один уровень сложности.

Для решения данных проблем используется программное обеспечение для совещаний. Этот вид программного обеспечения помогает избежать недопонимания и гарантирует, что все находятся на одной странице с актуальной информацией под рукой.

Целью исследования является изучение учёта и проведения корпоративных собраний на предприятии и создание на основе 1С:Предприятие системы, позволяющей их оптимизировать.

Для достижения поставленной цели, были выделены следующие задачи:

• Рассмотреть основные виды корпоративных собраний и процессы, связанные с ними;

• Определить основные требования к процессу проведения корпоративных собраний на предприятии;

• Разработка системы, автоматизирующей учёт и проведение корпоративных собраний на предприятии;

• Создание отчёта о проведённом научном исследовании.

Правильное программное обеспечение для управления собранием будет действовать, как эффективный инструмент учёта и процесса проведения собраний.

Внедрение системы автоматизации учёта и проведения собраний на предприятии позволяет компании автоматизировать следующие процессы, связанные с проведением собраний и совещаний:

• Планирование совещания;

• Согласование даты, места, времени, повестки;

• Автоматическая рассылка приглашений;

• Ликвидация рутинных операций по документальному оформлению результатов совещания (функции автоматической генерации файлов протоколов и отчётов по настраиваемым шаблонам);

• Оперативное информирование исполнителей (автоматическая генерация заданий по решениям, принятым на совещании).

Основной алгоритм системы можно реализовать с помощью системы 1С:Предприятие.

Для реализации требуемого функционала на 1С нам необходимо будет использовать справочники, документы и перечисления. Будет представлено несколько определённых типов собраний. Информация о каждом проведённом собрании будет собираться в документ. В итоге мы получаем готовую информационную систему, позволяющую оптимизировать контроль за проводимыми в организации собраниями и мероприятиями, следить за посещаемостью сотрудниками организации мероприятий, и, если потребуется, обратиться к истории проведённых мероприятий и получить нужную информацию о ранее прошедшем мероприятии.

В данной научно-исследовательской работе был проведён анализ выбранной предметной области, определена важность оптимизации учёта и проведения собраний на предприятии и её положительное влияние на рабочие процессы внутри компании. На основе статей от отечественных и зарубежных источников был сделан вывод об актуальности исследуемой темы.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Лященко Ю.В., Коржан И.А., Жереб Л.А. Проблемы и перспективы использования информационных технологий при проведении совещаний и заседаний // Актуальные проблемы авиации и космонавтики. 2014. №10. URL: https://cyberleninka.ru/article/n/problemy-i-perspektivy-ispolzovaniya-informatsionnyh-tehnologiy-pri-provedenii-soveschaniy-i-zasedaniy (дата обращения: 01.10.2022).

2. Laitinen, Kaisa & Valo, Maarit. (2018). Meanings of communication technology in virtual team meetings: Framing technology-related interaction. International Journal of Human-Computer Studies. 111. 12-22. 10.1016/j.ijhcs.2017.10.012.

**УДК 004.021**

**Артемов А.С.**

# ВЫБОР МЕТОДА МАШИННОГО ОБУЧЕНИЯ ДЛЯ ОЦЕНКИ СТОИМОСТИ НЕДВИЖИМОСТИ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В статье рассматривается выбор оптимального метода машинного обучения для наилучшей оценки стоимости недвижимости.

Оценка стоимости недвижимости – процесс определения рыночной стоимости объекта или отдельных прав в отношении оцениваемого объекта недвижимости. Оценка стоимости недвижимости включает: определение стоимости права собственности или иных прав, например, права аренды, права пользования и т.д. в отношении различных объектов недвижимости.

Оценочная практика показывает, что для выполнения отчёта об оценке стоимости квартиры специалисту требуется около часа. Автоматизация этого процесса позволит ускорить процесс принятия решения, учесть большее количество факторов оценки, снизить уровень субъективности оценки.

Рассмотрев принципиальную возможность применения экспертных систем, баз знаний, мультиагентных систем и машинного обучения для автоматизации определения рыночной стоимости недвижимости, выбор был сделан в пользу машинного обучения, которое позволяют учитывать неявные факторы формирования стоимости, адаптироваться к специфике территориальных рынков недвижимости [4].

Целью данной статьи является выбор наилучшего метода оценки стоимости недвижимости с использованием машинного обучения.

Для достижения указанной цели необходимо решить следующие задачи.

1. Предварительный отбор факторов, оказывающих влияние на рыночную стоимость недвижимости.

2. Подготовка обучающей и тестовой выборок.

Проведённые в статье исследования основаны на рыночных данных Нижнего Новгорода. При этом общие выводы, полученные в результате экспериментов, могут быть отнесены к другим регионам Российской Федерации. Для проведения экспериментов в рамках этой статьи были использованы данные реальных объявлений из категории «продажа вторичного жилья». Всего было собрано 9 050 объявлений [5].

Полученная выборка была разделена на две части: обучающую (70%) и тестовую (30%). Обучающая выборка использовалась для

тренировки моделей, а тестовая – для определения качества их предсказания [3].

Определим цены предложений, размещённых на сайтах о продаже жилой недвижимости, как $y$, предсказанные значения стоимости жилой недвижимости – как $\hat{y}$. Чтобы оценить эффективность используемой процедуры, для каждого результата рассчитываются следующие характеристики точности (метрики).

1. Коэффициент детерминации $R^2$ отражает долю объясняемой дисперсии модели. Чем ближе значение коэффициента детерминации к 1, тем сильнее соответствие модели данным:

$$R^2 = 1 - \frac{\sum_{i=1}^{n} \left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{n} \left(y_i - \bar{y}\right)^2}, \quad (1)$$

где $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$; $n$ – количество объектов в выборке.

2. Средняя абсолютная ошибка (mean absolute percentage error – $MAPE$) показывает, на сколько процентов в среднем ошибается модель:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100, \quad (2)$$

3. Медианная абсолютная ошибка (median absolute percentage error – $MedAPE$) отражает серединное значение среди всех упорядоченных значений процентных ошибок:

$$MedAPE = med\left\{ \left| \frac{y_i - \hat{y}_i}{y_i} \right| \right\} \times 100, \quad (3)$$

Указанные характеристики отдельно рассчитывались на выборке, используемой для обучения, и для выделенной тестовой выборки.

Точность оценки в большой степени зависит от набора признаков (ценообразующих параметров), с помощью которых идентифицируется объект оценки, поэтому важной частью формирования исходных данных является определение состава признаков для описания каждого объекта. Следует отметить, что полнота описания объекта ограничивается информацией об объектах, которые содержатся в объявлениях на продажу, размещенных на соответствующих ресурсах.

Содержательный анализ рынка жилья позволил выделить следующие существенные характеристики объекта, определяющие его рыночную стоимость:

1. Числовые переменные – год постройки, этажность, общая площадь квартиры, площадь кухни.

2. Категориальные переменные – район, количество комнат, материал стен, этаж размещения, территориальная зона.

Существует проблема, которую следует решить на этапе подготовки данных. Она связана с наличием в объявлениях «экстремальных» объектов (выбросы, ложные объявления с завышенными и заниженными ценами, с неадекватными и противоречивыми характеристиками объекта и т. п.). В связи с этим выборка, сформированная из рыночных данных, для исследований была подготовлена должным образом – удалены все «экстремальные» объекты. Все значения числовых признаков были приведены к единому типу, например, к целочисленному или к вещественному. Также все числовые характеристики были переведены в одинаковую разрядность. Категориальные признаки также требуют стандартизации – все буквы приводятся к строчным, чтобы «Кирпичные» стены не отличались от «кирпичных», убираются лишние пробелы и знаки препинания, исправляются орфографические ошибки, допущенные продавцом. Некоторые характеристики требуют более конкретных преобразований, например, значения признака «этаж размещения» объекта недвижимости были преобразованы в три вида: первый этаж, средний и последний.

Для большинства моделей входные данные должны передаваться в числовом представлении (за исключением CatBoost), поэтому к категориальным переменным был применен метод one-hot encoding, основной идей которого является замена одного признака, принимающего N значений на N бинарных признаков, принимающих значения 0 или 1 в зависимости от исходного значения [2].

Существует много методов машинного обучения, которые в принципе могут быть использованы для обеспечения процесса определения рыночной стоимости объекта недвижимости. В этом разделе приведены результаты исследований различных методов машинного обучения, в том числе широко распространенной в оценочной практике линейной регрессии, классических алгоритмов машинного обучения (random forest, gradient boosting), более современных моделей (xgboost, catboost). Кроме того, рассмотрены результаты оценки, основанные на применении нейронной сети. Таким образом, в работе рассматриваются следующие методы [1]:

1. Linear Regression.
2. Random Forest.
3. Gradient Boosting.
4. XGBoost.
5. CatBoost.
6. Neural Network.

Для адекватной оценки требуется достаточно полное описание объекта. Степень идентификации актива как объекта оценки определяется

полнотой его описания с помощью ценообразующих параметров. Причем существенную роль играет не только количество параметров, но и степень влияния каждого из них при оценке фактора на рыночную стоимость объекта. В таблице 1 показана значимость каждого из используемых признаков с точки зрения его влияния на стоимость объекта.

Таблица 1 – Значимость признаков с точки зрения его влияния на стоимость объекта

| Признак | Значимость признака |
|---|---|
| Район | 23,73 |
| Год постройки | 16,85 |
| Территориальная зона | 12,86 |
| Этажность дома | 10,52 |
| Общая площадь | 8,01 |
| Ремонт | 7,68 |
| Материал стен | 6,48 |
| Площадь кухни | 5,78 |
| Количество комнат | 5,03 |
| Этаж размещения | 3,06 |

Первые вычислительные эксперименты были проведены без учёта некоторых значимых факторов. При таком подходе местоположение объекта характеризовалось только районом города, и не было учтено физическое состояние жилья (результаты приведены в таблице 2 в столбцах с номером 1). Второй эксперимент был проведён на тех же данных, но с добавлением в качестве ценообразующего признака территориальной зоны (результаты приведены в таблице 2 в столбцах с номером 2). Третий эксперимент, в котором, наряду с учитывающимися раннее факторами, добавлен фактор, отражающий состояние объекта (результаты приведены в таблице 2 в столбцах с номером 3).

Таблица 2 – Результаты расчётов точности на тестовой выборке для разных методов и разных экспериментов

| Метод | $R^2$ | | | *MAPE* | | | *MedAPE* | | |
|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 1 | 2 | 3 | 1 | 2 | 3 |
| Linear Regression | 0,4 | 0,65 | 0,66 | 14,5 | 11,7 | 11,5 | 11,9 | 9,4 | 8,9 |
| Random Forest | 0,73 | 0,79 | 0,81 | 9,6 | 8,7 | 8,1 | 7,4 | 6,7 | 5,8 |
| Gradient Boosting | 0,69 | 0,74 | 0,78 | 11,1 | 10,2 | 8,6 | 8,9 | 8,3 | 6,4 |
| XGBoost | 0,70 | 0,73 | 0,78 | 10,3 | 9,8 | 9,0 | 8,1 | 7,9 | 6,9 |
| CatBoost | 0,73 | 0,77 | 0,77 | 10,3 | 9,2 | 9,1 | 8,2 | 7,2 | 7,1 |
| Neural Network | 0,60 | 0,71 | 0,71 | 11,4 | 9,5 | 9,4 | 8,1 | 7,2 | 7,1 |

Сопоставляя результаты расчётов, можно сделать следующие выводы:

1. Наилучшие результаты по всем показателям получены при использовании алгоритма Random Forest. Наихудшим образом сработал метод Linear Regression.

2. По мере добавления новых факторов медианная ошибка оценки (MedAPE) снижается.

3. При использовании обучающей выборки, хорошо подготовленной для обработки, медианная ошибка показывает вполне приемлемые результаты, близкие к лучшим результатам в мировой практике.

4. Ещё раз подтвердилось, что наиболее значимым ценообразующим фактором для жилой недвижимости является расположение объекта.

В статье была рассмотрена проблема определения стоимости жилой недвижимости по её характеристикам. Для решения этой задачи были использованы различные методы машинного обучения и интеллектуального анализа данных.

В ходе исследований были собраны и проанализированы данные, размещённые на сайтах о продаже жилой недвижимости, обучено несколько алгоритмов машинного обучения, выбран лучший метод (Random Forest). Результаты исследований, полученные в рамках этой работы, подтверждают эффективное применение машинного обучения для решения задачи определения стоимости объектов жилой недвижимости.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Шолле Ф. Глубокое обучение на Python. - СПб.:Питер, 2018. - 400 с.

2. Turing A.M. Computing Machinery and Intelligence // Mind. - 1950. - Vol. 59, No. 236. - P. 433-460.

3. Рашка С. Python и машинное обучение. - М.: ДМК-Пресс, 2017. - 420 с.

4. Cyberleninka [сайт] – URL: https://cyberleninka.ru/article/n/ispolzovanie-neyronnyh-setey-dlya-otsenki-rynochnoy-stoimosti-nedvizhimosti (дата обращения: 15.11.2022).

5. Cyberleninka [сайт] – URL: https://cyberleninka.ru/article/n/massovaya-otsenka-obektov-nedvizhimosti-na-osnove-tehnologiy-mashinnogo-obucheniya-analiz-tochnosti-razlichnyh-metodov-na-primere (дата обращения: 22.11.2022).

**УДК 004.891.2**

**Артёмов Я.А.**

# ИССЛЕДОВАНИЕ АЛГОРИТМОВ ИНТЕЛЛЕКТУАЛЬНОЙ ОБРАБОТКИ ТЕКСТОВОЙ ИНФОРМАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», г. Рязань

В статье рассматриваются существующие подходы для обработки текстовой информации.

В настоящее время количество информации, производимой человечеством, растёт в экспоненциальной прогрессии и в мире наблюдается потребность в новых методах анализа информации разного вида, поскольку существующие алгоритмы уже не могут справиться с требуемым объёмом информации. Актуальность научного исследования обусловлена необходимостью обработки огромного количества текстовой информации.

Целью научного исследования является изучение существующих методов и их практического применения к обработке текстовой информации.

Исходя из цели были сформулированы следующие задачи: изучить методы и существующие программные средства интеллектуальной обработки текстовой информации.

В ходе работы над данной статьёй были применены такие теоретические методы исследования как: анализ, классификация, аналогия, абстрагирование и моделирование.

Как же можно анализировать текстовую информацию? Существует множество подходов. Для начала нам необходимо подготовить данные для анализа, ведь компьютер не может анализировать слова в исходном виде. Наиболее простым методом кодирования слов является построение вектора каждого слова методом one-hot encoding. Для этого нам нужно создать нулевой вектор для словаря языка и поставить единицу только в элементе с индексом, соответствующим нужному слову. Самый простой способ анализа текста состоит в сложении всех векторов слов, которые включает текст. Так мы получим вектор частоты употребления слов в тексте. Но мы теряем всю смысловую нагрузку и данный метод не подходит для анализа больших текстов. Было предложено составлять матрицы из пар всех слов языка, но возникла проблема хранения и анализа таких матриц.

В 2013 году Томаш Миколов предложил иной подход к анализу слов word2vec. Его подход основан на гипотезе, которую в науке принято называть гипотезой локальности — «слова, которые встречаются в одинаковых окружениях, имеют близкие значения». Близость в данном случае понимается очень широко, как то, что рядом могут стоять только

сочетающиеся слова. Например, для нас привычно словосочетание «заводной будильник». А сказать «заводной апельсин» мы не можем — эти слова не сочетаются. Таким образом, имея данные о близости слов можно предсказать следующее слово. При помощи усреднения векторов окружающих слов можно найти вектор, близкий вектору слова, которое будет идти дальше. Усреднение может быть выполнено при помощи функции косинусного расстояния окружающих векторов. Данный метод обработки может быть реализован при помощи простой трёхслойной нейронной сети, состоящей из входного слоя (Input), скрытого слоя (Projection) и выходного слоя (Output).

Схема реализации этой модели представлена на рисунке 1.

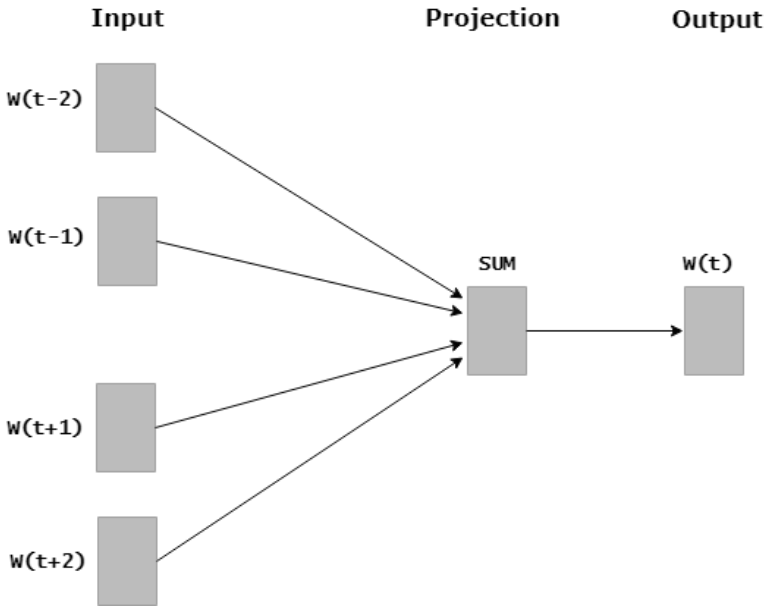


Рисунок 1 — Схема реализации модели CBOW

Также Миколовым сразу был предложен прямо противоположный подход, который он назвал Skip-Gram, то есть «словосочетание с пропуском». Мы пытаемся из данного нам слова угадать его контекст (точнее вектор контекста). Данная модель является обратным представлением ранее рассмотренной модели CBOW и также может быть реализована при помощи простой трёхслойной нейронной сети, состоящей из входного слоя (Input), скрытого слоя (Projection) и выходного слоя (Output). Схема реализации этой модели представлена на рисунке 2.

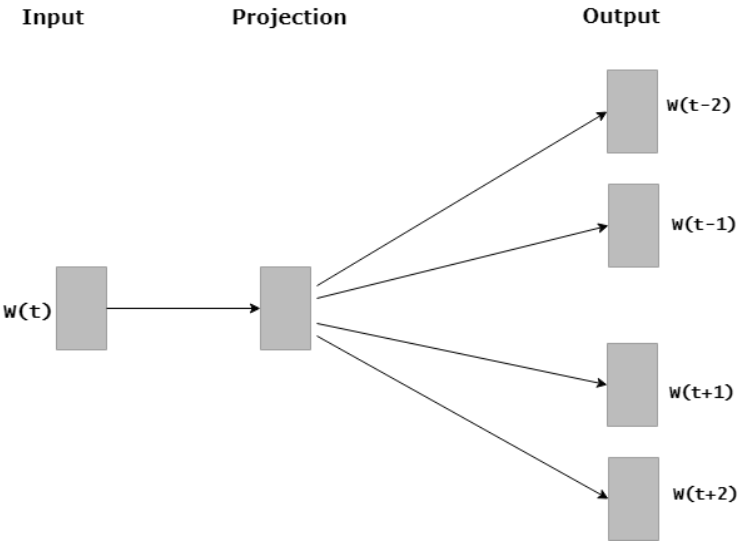


Рисунок 2 — Схема реализации модели Skip-Gram

Учёными были предложены методы улучшения качества модели нейронной сети, работающей по алгоритму предсказания слов Миколова. Одним из методов повышения качества нейронных сетей является применение дополнительных функций оптимизации. Для данной сети чаще всего применяется дивергенция Кульбака-Лейблера. Оказалось, что оптимизировать эту формулу достаточно сложно. Прежде всего из-за того, что она рассчитывается с помощью функции softmax по всему словарю. Поскольку многие слова вместе не встречаются, большая часть вычислений в softmax является избыточной. Был предложен элегантный обходной путь, который получил название Negative Sampling. Суть этого подхода заключается в том, что мы максимизируем вероятность встречи для нужного слова в типичном контексте и одновременно минимизируем вероятность встречи в нетипичном контексте. Также был предложен другой подход для улучшения качества сети — можно не менять исходную формулу, а попробовать посчитать сам softmax более эффективно. Например, используя бинарное дерево Хаффмана, построенное по всем словам в словаре. При использовании простого softmax для подсчета вероятности слова, приходилось вычислять нормирующую сумму по всем словам из словаря, требовалось N(V) операций. Теперь же вероятность слова можно вычислить при помощи последовательных вычислений, которые требуют N(log(V)).

Описанные подходы являются наиболее распространёнными в настоящее время. В ходе работы над статьёй была реализована методика анализа текста и предсказания слов, предложенная Томашом Миколовым при помощи нейронной сети. Нейронная сеть была создана при помощи средств языка программирования Python и библиотек nltk и gensim. Процесс тренировки устроен следующим образом: мы берем последовательно (2k+1) слов, слово в центре является тем словом, которое должно быть предсказано. А окружающие слова являются контекстом длины по k с каждой стороны. Каждому слову в нашей модели сопоставлен уникальный вектор, который мы меняем в процессе обучения нашей модели. В ходе работы над статьёй была смоделирована и обучена нейронная сеть для предсказания слова по окружающим его словам. Таким образом, в ходе работы над статьёй были созданы и обучены 2 нейронные сети, реализующие модели CBOW и Skip-Gram. Был проведён эксперимент по сравнению данных нейронных сетей. Сравнение производилось следующим образом. Был выбран отрывок из книги «Алиса в стране чудес». Отрывок был передан нейронным сетям для анализа, после чего было решено сравнить косинусное сходство между парами слов из книги. Результаты обработки:

Косинусное сходство между 'Алиса' и 'Чудеса'- CBOW: 0.924929.

Косинусное сходство между 'Алиса' и 'Механизм' - CBOW: 0.374911.

Косинусное сходство между 'Алиса' и 'Чудеса' - Skip-Gram: 0.885471.

Косинусное сходство между 'Алиса' и 'Механизм' - Skip-Gram: 0,256892.

Как видно из результатов исследования, слова 'Алиса' и 'Чудеса' имеют высокое косинусное сходство, а слова 'Алиса и 'Механизм' имеют низкое косинусное сходство, что подтверждается их близким употреблением в рассматриваемом тексте. Кроме того, можно заметить, что рассмотренные нейронные сети дают схожие результаты. Таким образом можно анализировать разные сочетания слов. Рассмотренные модели нейронных сетей также можно улучшить, что приведёт к повышению качества анализа текстовой информации.

В ходе работы над статьёй были изучены методы обработки текстовой информации, способы практического применения изученных методов, а также созданы 2 нейронные сети, реализующие 2 модели обработки текстовой информации, после чего было проведено сравнение созданных нейронных сетей. Полученные знания будут применены для улучшения качества работы созданных нейронной сети с целью повышения их эффективности.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1.Collobert R. et al. Natural language processing (almost) from scratch //Journal of Machine Learning Research. – 2011. – Т. 12. – №. Aug. – С. 2493-2537

2.Trofimov I. V. Person name recognition in news articles based on the persons-1000/1111-F collections //16th All-Russian Scientific Conference Digital Libraries: Advanced Methods and Technologies, Digital Collection, RCDL. – 2014. – С. 217- 221.

3. Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. CoRR, abs/1301.3781,

**УДК 004.32.26**

**Александров В.В., Цуканова Н.И., Головкин Н.В., Шурыгина О.В.**

# ИССЛЕДОВАНИЕ СПОСОБОВ ПОВЫШЕНИЯ ТОЧНОСТИ КЛАССИФИКАЦИИ ФРАГМЕНТОВ КОЛЛЕКТИВНОГО ДОГОВОРА

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

Рассматриваются вопросы повышения точности классификации фрагментов коллективного договора с помощью нейронных сетей. Приводятся результаты экспериментального исследования алгоритмов классификации.

Коллективный договор (КД) организации заключается между работниками и работодателем в лице их представителей и является правовым актом, предназначенным для дополнительного регулирования социально-трудовых отношений в данной организации.

В работе [1] рассматривается реализация количественного метода оценки правовой эффективности КД, характеризуемой значением количественного показателя Эф. Такой показатель, а также сама методика, позволяют сравнивать коллективные договоры разных учреждений между собой и определять организации, в которых созданы более благоприятные условия труда для работников, а также проводить анализ причин неэффективности отдельных КД.

В работе [2] описаны этапы предложенной в [1] количественной оценки эффективности КД. Среди этих этапов наиболее важными и трудными являются 2- й и 3- й этапы. На этих этапах для каждого фрагмента КД должны быть определены 3 характеристики: «Раздел», «Вопрос» и «Качество». «Раздел» и «Вопрос» характеризуют правовой смысл каждого фрагмента, т. е. показывают, какие социально-трудовые параметры или правовые нормы фрагмент отражает. «Раздел» указывает на принадлежность фрагмента к определённому разделу трудового права, например: Оплата труда или Рабочее время, Время отдыха. Всего 11 разделов. «Вопрос» уточняет, какой вопрос внутри раздела определён во

фрагменте, например, Даты выплаты заработной платы, Гарантии оплаты, Нормы учебной нагрузки, Перечень дополнительных отпусков. Для каждого раздела свой список вопросов. С учетом этих двух характеристик фрагменты могут принадлежать 80 разным классам.

Третья характеристика «Качество» отражает, насколько в новом трудовом договоре условия труда и его оплаты, охраны труда, социально-бытовые условия, льготы и гарантии более благоприятные по сравнению с тем, что установлены в трудовом законодательстве. Оценить значение этой характеристики наиболее сложно.

Процесс разметки фрагментов является задачей классификации. Он очень трудоёмкий, требует наличия опытных экспертов и автоматизации.

В работе [2] предложена и показана целесообразность использования нейронных сетей (НС) для реализации 2-го этапа – оценки фрагмента по характеристикам «Раздел» и «Вопрос». Позже эта же методика была распространена и на оценку «Качества».

Суть методики в следующем. Для оценки каждой характеристики создаётся и обучается классификатор: классификатор по разделу, классификатор по вопросу, классификатор по качеству. Все классификаторы – это ансамбли двух нейронных сетей. Для их обучения используется набор данных (выборка), состоящий из фрагментов коллективных договоров прошлых лет. Выборка была получена из базы данных коллективных договоров Вузов России ведомственной лаборатории РГРТУ по автоматизированному контролю, анализу и оценке эффективности коллективно-договорных актов в сфере образования.

Каждый фрагмент имеет следующие характеристики (атрибуты): «Текст», «Код», «Раздел», «Вопрос» и «Качество». Все они заданы для каждого фрагмента выборки и могут рассматриваться как метки классов. Раздел и вопрос тесно связаны между собой, что отражается в атрибуте фрагмента «Код». Каждое значение Кода однозначно соответствует паре «Раздел-Вопрос». Выборка сохранена в файле Excel, что позволяет легко провести её анализ. Анализ показал, что классы в выборке представлены неравномерно.

При обучении нейронных сетей на их вход подается текст фрагмента (атрибут «Текст»), в качестве эталона выхода выступают метки фрагмента по соответствующей характеристике: для классификатора по разделу «Раздел», для классификатора по вопросу «Вопрос», для классификатора по Коду «Код» и т.д.

Для реализации нейронных сетей были выбраны следующие инструментальные средства: язык Python, библиотеки TensorFlow, Keras, виртуальная среда Google Colab [2].

Точность оценки нового фрагмента зависит от точности каждого классификатора и их совместной работы. Поэтому важной задачей

является добиться как можно большей точности от каждого классификатора. В настоящей работе приводятся результаты экспериментального исследования различных методов повышения точности классификации фрагментов Коллективного договора.

Сначала повышение точности классификаторов осуществлялось за счет выбора значений гиперпараметров процесса обучения НС, таких как объем выборки, размер словаря, длина фрагмента. В таблице 1 приведены результаты такого исследования.

Таблица 1 – Результаты исследования

| **Размер словаря Кл. «По разделу»** | | **Обьем выборки Кл. «По разделу»** | |
|---|---|---|---|
| *Размер* | *Точность* | *Обьем* | *Точность* |
| 1000 | 84% | 46894 | 87% |
| 5000 | 86,84% | 108241 | 90% |
| 7000 | 87,9% | | |
| 10000 | 88,34 | | |

Затем были разработаны разные архитектуры системы классификаторов и проведена оценка их качества по точности.

Система классификаторов основана на следующих элементах: 1) линейный классификатор по Коду; 2) иерархический классификатор сначала по Разделу, затем по Вопросу внутри полученного Раздела; 3) разные варианты классификаторов по Качеству.

Т.к. в иерархическом классификаторе [2] классификация по вопросам зависит от раздела, то для каждого раздела нужен свой классификатор. Следовательно, необходимо иметь столько классификаторов (нейронных сетей) по вопросам, сколько разделов.

Исследование выборки показало значительную неравномерность её по всем характеристикам: разделу, особенно по вопросам и качеству. Поэтому решено было при обучении любого классификатора сначала сбалансировать выборку и только потом обучать. Как повлиял процесс балансировки классов выборки на точность классификации показано в таблице 2.

Таблица 2 – Влияние точности классификации

| | **Кл. по разделам** | **Кл. по вопросам** | **Кл. по Коду** | **Кл. по Качеству** |
|---|---|---|---|---|
| **Без балансировки** | 88,5% | 82% | 60% | 78% |
| **С балансировкой** | 92% | 77%-90% | 82% | 83% |

Наиболее интересен результат для классификатора по Коду. Если без балансировки выборки по классам его использование было нецелесообразно из-за низкой точности 60%, то выравнивание выборки

позволило получить результат сравнимый или даже немного лучший, чем с иерархическим классификатором. Иерархический классификатор сложен, требует больших затрат на обучение и сохранение большого числа моделей для каждого раздела. В то же время значение Кода дает однозначный ответ, к какому разделу и вопросу относится фрагмент. Следовательно, при балансировке выборки следует отдать предпочтение классификатору по Коду.

Из анализа полученных результатов следует:

1) лучшими значениями гиперпараметров процесса обучения являются: объем выборки = 108241; размер словаря = 10000; длина фрагмента = 250; 2) перед обучением должна быть проведена балансировка выборки по классам; 3) при использовании балансировки выборки лучшим и более экономичным оказался линейный классификатор по Коду, использование которого позволяет получить пару Раздел-Вопрос; 4) самая низкая точность у классификатора по Качеству, поэтому необходимо искать другую методику оценки этой характеристики

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1.Александров В.В., Макаров Н.П., Шустов А.С. Автоматизированный анализ и оценка статей коллективных договоров //Вестник Рязанского государственного радиотехнического университета. – 2013. – № 3 (45). – С. 71-75.

2. Александров В.В., Цуканова Н.И., Головкин Н.В., Шурыгина О.В. Методы интеллектуального анализа данных и машинного обучения при автоматизированной оценке эффективности коллективных договоров //Математическое и программное обеспечение вычислительных систем: Межвуз.сб.науч.тр./ Под ред. Г.И.Овечкина – Рязань: Изд. ИП Коняхин А.В. июнь 2022.-92с.

**УДК 004.056.5**

**Бобылева Е.В., Хорева А.А.**

# ИССЛЕДОВАНИЕ АЛГОРИТМОВ В ОБЛАСТИ РАЗРАБОТКИ CRM СИСТЕМ ДЛЯ СТУДИИ ТАНЦЕВ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

Целью данной работы является исследование CRM-систем для студии танцев и алгоритмов по их разработки.

Под понятием CRM подразумевают систему управления взаимоотношениями с клиентами. В её состав входит огромное число инструментов, нацеленных на повышение критериев и показателей

эффективности деятельности как сотрудников по отдельности, так и всей компании в целом.

Наиболее важным элементом архитектуры CRM системы является структурно-функциональная организация, в процессе которой происходит сбор, анализ и обработка информации. Архитектура практически любой CRM-системы представляет собой базу данных, в которой мы ведем учёт всех клиентов [1].

Общая схема использования CRM-систем для построения взаимоотношений с клиентами представлена на рисунке 1.


Рисунок 1 – Общая схема использования CRM-систем для построения взаимоотношений с клиентами

К недостаткам существующих CRM-систем для студий танцев и аналогичных предприятий относится отсутствие автоматизированного составления расписания. В большинстве CRM-систем реализовано заполнение расписания, а не его составление.

Поскольку значимой частью CRM системы будет являться составление расписания занятий для студии танцев, то исследуем алгоритмы и методы по автоматизации составления расписания.

В случае расписания для студии танцев данная задача больше подходит к задачам о назначениях, которая представляет собой частный случай транспортной задачи. Поэтому для её решения можно воспользоваться любым алгоритмом решения задач линейного программирования. Если число исполнителей и число выполняемых работ совпадают, то задача является сбалансированной, в противном случае - несбалансированной [6].

Для нахождения оптимального решения задачи с простейшей линейной моделью обычно применяют стандартные алгоритмы,

учитывающие специфику её ограничений и целевой функции, например, венгерский метод и метод Мака [4].

**Венгерский метод.** Специфические особенности задач о назначениях послужили поводом к появлению эффективного венгерского метода их решения. Основная идея венгерского метода заключается в переходе от исходной квадратной матрицы стоимости С к эквивалентной ей матрице Сэ с неотрицательными элементами и системой n независимых нулей, из которых никакие два не принадлежат одной и той же строке или одному и тому же столбцу.

**Метод Мака.** Метод Мака основан на идее выбора в каждой строке минимального элемента. Однако минимальные элементы строк не распределены по всем n столбцам матрицы. Здесь используется идея сложения (или вычитания) одного и того же значения со всеми элементами строки или столбца, чтобы распределить минимальные элементы строк по столбцам (тогда они образуют оптимальный выбор).

**Алгоритм Литтла.** Данный алгоритм является частным случаем применения метода «ветвей и границ» и относится к числу точных алгоритмов, однако при решении задач большой размерности, когда разброс данных невелик, этот алгоритм сходится, но очень медленно. Общая идея заключается в том, чтобы разделить огромное число перебираемых вариантов на классы и получить оценки для этих классов, чтобы иметь возможность отбрасывать варианты не по одному, а целыми классами. Трудность состоит в том, чтобы найти такое разделение на классы (ветви) и такие оценки (границы), чтобы процедура была эффективной[7].

**Динамическое программирование**. Для решения сбалансированной задачи о назначениях можно предложить следующий простой алгоритм: на каждом шаге назначений, начиная с первого исполнителя, выбирать самого эффективного. Однако такой способ решения, как правило, не приводит к получению оптимального решения. Если на первых шагах выбирать самых эффективных (неэффективных) исполнителей, то на последних шагах алгоритма приходится выполнять назначения, которые могут внести негативный вклад в суммарный критерий.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Захарова Т.И., Иванов А.А., Кокоулина О.П., Халиль И.А., Цимбалюк А.А. Сущность концепции управления взаимоотношениями с клиентами (crm-системы) и её роль в повышении эффективности деятельности в современных организациях // Инновации и инвестиции. 2021. №6.

2. Богданова Е.Л. Оптимизация в проектном менеджменте: линейное программирование: учебное пособие / Е.Л. Богданова, К.А. Соловейчик, К.Г. Аркина. – СПб.: Университет ИТМО, 2017. – 165 с.

3. Шевченко А.С. Методы оптимизации. Линейное программирование: учебно-методическое пособие – Рубцовск: Рубцовский институт (филиал) АлтГУ, 2016. – 162 с.

4. Балашова И.Ю. Модели и алгоритмы решения обобщенных задач о назначения : дис. … канд.тех.наук: 09.04.02 – Пенза, 2020 – 97 с.

5. A.Ahmed, Afaq Ahmad, A new method for finding an optimal solution of assignment problem // International Journal of Modern Mathematical Sciences, 2014. – 10 - 15 с.

6. Щукина Н.А. Некоторые подходы к решению задачи о назначениях // Проблемы экономики и менеджмента. 2016. №5 (57).

7. Россолов С. Ю. Применение метода Мака в решении задач о назначениях – СПб, 2017.

**УДК 004.056.5**

**Буланова И.А., Попов Д.И., Пылькин А.Н.**

# ПОСТРОЕНИЕ СЕМАНТИЧЕСКОЙ МОДЕЛИ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ ВЫСШЕГО ОБРАЗОВАНИЯ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань, Сочинский государственный университет, Сочи

Целью данной работы является описание и реализация алгоритма построения семантической модели рабочей программы учебной дисциплины высшего образования с использованием библиотеки для построения графов Graphiz.

Для выполнения формализации предметной области часто применяются методы с использованием онтологий. Онтология строится как сеть, состоящая из концептов предметной области и связей между ними.

Онтологии используются различными способами в образовательных системах, в зависимости от задачи, которую они решают.

Во-первых, они используются для моделирования и управления образовательной программой. Образовательная программа и учебный план представляется в виде онтологии, что облегчает задачи управления, анализа и оценки программы. Также такое использование онтологии позволяет разработчикам образовательных программ улучшать их путем определения основных элементов образовательной программы, связывания одних учебных единиц с другими. Во-вторых, онтологии могут использоваться для контроля понятийных знаний учащихся. В-третьих, с помощью онтологий можно описать предметные области различных дисциплин и задач обучения, что является наиболее популярным способом использования онтологий [1].

Существуют различные разработанные онтологические методы структуризации учебного контента, которые успешно внедрены в системы электронного обучения [2-4].

Закономерным продолжением онтологического подхода к структуризации учебного курса является визуальное отображение разработанной онтологической модели в виде графа, для чего могут быть использованы семантические сети.

Для построения графов существуют различные библиотеки. Как наиболее подходящая для реализации задачи была выбрана библиотека Graphviz. Библиотека Graphviz позволяет обрабатывать файлы на языке DOT и имеет различные алгоритмы размещения графов. Среди алгоритмов размещения был выбран алгоритм Сугиямы, или алгоритм послойного рисования графа.

Ниже приведена блок-схема обобщённого алгоритма построения графа с использованием библиотеки Graphiz (рисунок 1).

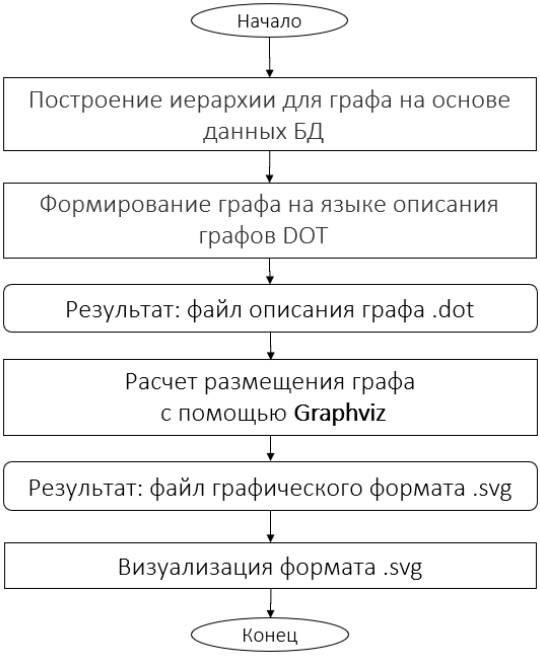


Рисунок 1 – Обобщенный алгоритм построения графа семантической модели рабочей программы учебной дисциплины

Данные из БД, используемые в графе, обрабатываются рекурсивными процедурами, после чего формируется файл на языке описания DOT. Затем Graphiz принимает файл .dot и рассчитывает

расположение объектов графа и отображает его в формате svg или других форматах. Был выбран формат svg, так как он является векторным графическим форматом и доступен к просмотру в браузере.

Ниже представлен интерфейс приложения по работе с графами (рисунок 2). Приложение позволяет строить граф, корень которого выбирается пользователем. Граф можно обойти как слева направо, так и сверху вниз.

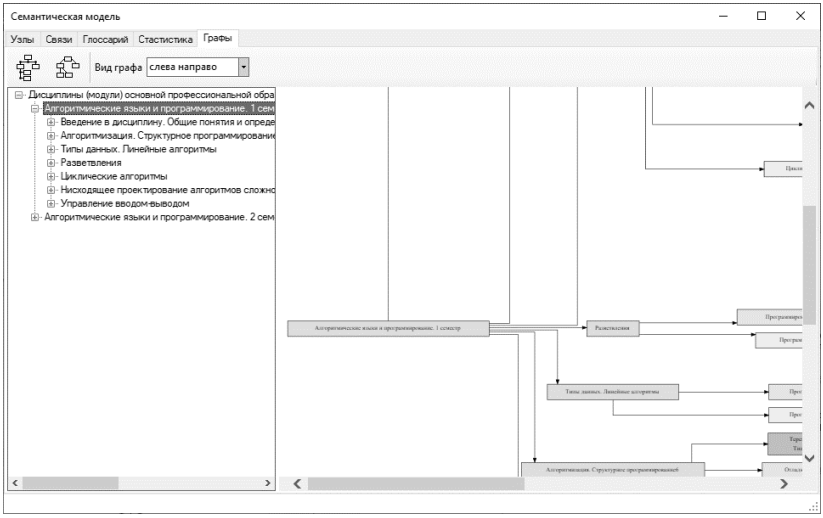


Рисунок 2 – Интерфейс приложения

Ниже представлен пример графа одного из разделов дисциплины (рисунок 3). Он состоит из трёх лекций, двух практических занятий и самостоятельной работы. Лекции содержат понятия, которые могут включаться и в другие лекции. Цвета элементов могут быть настроены пользователем.

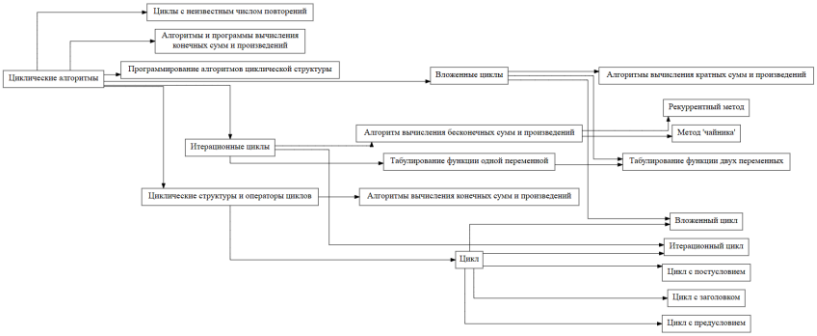


Рисунок 3 – Пример графа раздела дисциплины

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Смирнова Е.В., Добрица Е.К., Демиденко Н.О. Использование онтологий в образовательных процессах // Проблемы современной науки и образования. – Иваново, 2017. - №22. - С.70-74.

2. Ручкин В.Н., Фулин В.А. Использование онтологического метода структуризации учебного контента // Известия ТулГУ. Технические науки. – Тула, 2014. - №6. – С.168-174.

3. Зеленко Л.С., Шумская Е.А. Разработка онтологической модели учебного курса для систем электронного дистанционного обучения // Программные продукты и системы. – Тверь, 2018. - №1. - С.56-59.

4. Попов Д.В., Макушкина Л.А. Исследование методов построения конвертера онтологических моделей курса // Современные научные исследования и инновации. – Москва, 2014. - № 1. – С.4-7.

5. Каширин, Д.И. Полиморфическое представление знаний в Semantic Web: Монография [Текст] / Д.И. Каширин, И.Ю. Каширин, А.Н. Пылькин. — М.: Горячая линия-Телеком, 2009.- 136 с.

6. Бобылева Е.В. Семантическая модель рабочей программы учебной дисциплины / Бобылева Е.В., Буланова И.А., Пылькин А.Н. // Современные технологии в науке и образовании – СТНО-2022 [Текст]: сб. тр. V междунар. науч.-техн. форума: в 10 т. Т.4./ под общ. ред. О.В. Миловзорова. – Рязань: Рязан. гос. радиотехн. ун-т, 2022. – С.61-64.


**УДК 004.056.5**

**Буланова И.А., Швечкова О.Г., Бобылева Е.В.**

# ИССЛЕДОВАНИЕ АЛГОРИТМОВ СТЕГАНОГРАФИИ В РАМКАХ ДИСЦИПЛИНЫ «ЗАЩИТА ИНФОРМАЦИИ»

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

Целью данной работы является описание возможного подхода к изучению алгоритмов стеганографии как одного из видов криптографических методов защиты информации в рамках учебной дисциплины «Защита информации».

Актуальность темы исследования определяется лавинообразным ростом попыток несанкционированного доступа к информации ограниченного использования, предназначенной для определённого круга лиц, в том числе к персональным данным гражданина. В данном случае, особенно при передаче данных по сетям, наиболее эффективным методом предотвращения попыток различного рода вредоносных воздействий принято считать криптографические методы защиты информации.

Рассматривая принятую в научных кругах классификацию криптографических методов защиты информации, следует отметить

стоящий особняком подход к обеспечению секретной передачи или хранению данных – стеганографию.

Стеганография – особый способ передачи или хранения информации путём специфической формы преобразования с целью сохранения в тайне самого факта такой передачи или хранения.

В отличие от криптографического преобразования, которое модифицирует содержимое тайного сообщения, стеганография предполагает засекречивание самого факта возможного сокрытия информации. Стеганографические преобразования, как правило, используют совместно с методами криптографии, таким образом дополняя её возможности.

Преимущество стеганографии над непосредственно методами криптографии в том, что сообщения, преобразованные данными алгоритмами, не выглядят модифицированными, тем самым не привлекают к себе внимания. Таким образом, криптография защищает содержание сообщения, а стеганография защищает сам факт наличия каких-либо скрытых посланий.

Хотя стеганография как способ сокрытия секретных данных известна уже на протяжении тысячелетий, компьютерная стеганография - молодое и развивающееся направление, поэтому с полным правом может считаться новой информационной технологией в сфере защиты информации.

Компьютерная стеганография как новая часть научного направления информационной безопасности окончательно была оформлена в рамках конференции Information Hiding: First Information Workshop в 1996 году, где были предложены единая терминология и основные термины. С данного времени используется понятие стеганографической системы или стегосистемы как совокупности методов и средств, которые используются для формирования скрытого канала передачи информации. Для обозначения скрываемой информации принят термин «сообщение», под которым может пониматься текст, изображение или аудиоданные.

В рамках изучения практического построения и программной реализации алгоритмов стеганографического преобразования информации предполагается выполнение лабораторного практикума, состоящего из двух лабораторных работ.

Теоретическая часть практикума включает в себя изучение следующих этапов проектирования стеганографических алгоритмов.

1. Архитектура (структурная схема) стегосистем.

2.Требования к стегосистемам, включая свойства контейнера, стегосообщения.

3. Использование кода с исправлением ошибки.

4. Повышение надёжности встраиваемого сообщения возможностью дублирования.

5. Особенности реализации основных алгоритмов, таких как, например, работающие с самим цифровым сигналом, со «впаиванием» скрытой информации, связанные с особенностями форматов файлов.

Практическая часть лабораторных работ содержит следующие основные этапы компьютерной стеганографии.

1. Выбор информационного файла, т.е. скрываемого сообщения.

2. Выбор файла-контейнера, используемого для сокрытия информации. Его ещё называют файлом-контейнером.

3. Выбор стеганографической программы. В интернете имеется большое количество платных и бесплатных стеганографических программ.

4. Кодирование файла с установкой защиты нового файла по паролю.

5. Отправление сокрытого сообщения по электронной почте и его декодирование.

Первая лабораторная работа подразумевает освоение метода стеганографического преобразования информации на базе известного алгоритма LSB (Least Significant Bit, наименьший значащий бит (НЗБ)), суть которого заключается в замене последних значащих битов в контейнере (изображения, аудио или видеозаписи) на биты скрываемого сообщения. Разница между пустым и заполненным контейнерами должна быть неощутима для органов восприятия человека. Методы LSB являются неустойчивыми ко всем видам атак и могут быть использованы только при отсутствии шума в канале передачи данных.

Вторая лабораторная работа посвящена изучению методов текстовой стеганографии, которые подразделяются на синтаксические, которые не затрагивают семантику текстового сообщения, и лингвистические методы, которые основаны на эквивалентной трансформации текстовых файлов, сохраняющей смысловое содержание текста, его семантику, направлена приобретение навыков стеганографического сокрытия информации с использованием битов цветовой палитры изображений.

Изложенные теоретические методы стеганографической защиты информации в современном мире имеют особую практическую значимость, суть которой: незаметная передача информации, скрытое хранение информации, недекларированное хранение информации, защита исключительного права на базе цифровых отпечатков, защита авторского права, защита подлинности документов, индивидуальный отпечаток в системах электронного документооборота (СЭДО), водяной знак в DLP системах, подтверждение достоверности переданной информации и др.

Выполнение данных лабораторных работ предполагает наличие должных инженерных навыков студентов завершающей стадии получения степени бакалавра и является итогом изучения алгоритмов стеганографии как элемента инновационного механизма криптографической защиты информации.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Алгоритмы стеганографической защиты информации [Текст]: методические указания к лабораторным работам/ Рязан. гос. радиотехн. ун-т; сост.: О.Г. Швечкова. - Рязань, 2017. - 32 с.

2. Урбанович, П. П. Защита информации методами криптографии, стеганографии и обфускации [Текст]: учеб.-метод. пособие – Минск: БГТУ, 2016. – 220 с.

3. «Модификация стеганографического метода LSB для повышения секретности передачи сообщения»: А.В. Гиголаев, Н.А. Тярт, О.Г. Швечкова. «Современные технологии в науке и образовании СТНО-2018» в 11 т. Т. 4- Рязань, 2018. - стр. 43-47 с.

**УДК 004.421**

**Волков А.А.**

# РАЗРАБОТКА РАСПРЕДЕЛЕННОЙ ИНФОРМАЦИОННОЙ ОБРАЗОВАТЕЛЬНОЙ СИСТЕМЫ С МИКРОСЕРВИСНОЙ АРХИТЕКТУРОЙ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В статье рассмотрены информационно-образовательные системы, стандарты электронного образования, предложена архитектура для разрабатываемого приложения.

В настоящее время ресурсы Интернета стали общедоступными, что в свою очередь привело к повсеместному использованию электронного обучения. Электронное обучение (ЭО) – это система обучения, предполагающая использование информационно-коммуникационных технологий, а также мультимедийных материалов. Важной отличительной особенностью электронного обучения от традиционного является возможность выбора различных образовательных ресурсов исходя из индивидуальных особенностей и потребностей пользователя. Существует два вида систем электронного обучения: системы управления обучением (Learning Management Systems – LMS) и системы управления учебным контентом (Learning Content Management Systems – LCMS). Для разработки архитектуры приложения была выбрана система управления обучением – LMS.

Цель работы – изучение микросервисной архитектуры.

Поставленные задачи:

1. Рассмотреть основные элементы системы управления обучением.

2. Разработать микросервисную архитектуру системы управления обучением.

LMS включает в себя следующие компоненты: учебные материалы, обратная связь, средства коммуникации, распределение доступа, учёт деятельности, регистрация/авторизация пользователей. В электронном образовании выделяют следующие стандарты и форматы:

1. AICC – первый eLearning-стандарт. Стандарт aicc неоднократно дорабатывался и обновлялся, но ему становилось все тяжелее соответствовать требованиям технического прогресса. Тем не менее именно благодаря aicc разработчики LMS смогли адаптировать видеоконтент под мобильные устройства. Некоторые из приёмов aicc до сих пор используются в современных платформах. На данный момент считается устаревшим.

2. SCORM (1.2 и 2004) – универсальный стандарт электронного обучения. Формат представляет собой набор стандартов и спецификаций для LMS. Это улучшенная версия стандарта aicc, и на данный момент это самый популярный универсальный стандарт создания онлайн-контента, который до сих пор поддерживается большинством LMS. До внедрения этого формата было довольно сложно интегрировать курсы с системой обучения, если содержание не было адаптировано к конкретной платформе.

3. Tin Can (xApi) - улучшенный Scorm-стандарт. xApi – это набор спецификаций, при котором обучающие системы имеют возможность коммуницировать между собой, отслеживая и фиксируя учебные занятия. Tin Can значительно расширил количество информации о процессе обучения, которые обучающий курс передает в СДО. Главное его преимущество в использовании функции Learning Record Store (LRS – хранилище учебных записей): данные об активности учащегося хранятся в архиве LRS и передаются в LMS при выходе в Интернет, при этом весь прогресс обучения сохраняется.

4. Cmi 5 - это надстройка над xAPI. При реализации обязывает использовать параметры курса, которые были использованы в стандарте SCORM.

Для разрабатываемой LMS был выбран стандарт cmi5. Данный стандарт вводит необходимость в разработке новой системы – Learning Record Store (LRS), которая должна сохранять все активности от обучающихся. LRS может быть отдельным модулем внутри LMS или автономной системой. В контексте разработки приложения с микросервисной архитектурой наиболее предпочтительный вариант выделения данной системы из LMS в отдельный независимый сервис.

На основе собранной информации была разработана архитектура будущего приложения, представленная на рисунке 1.

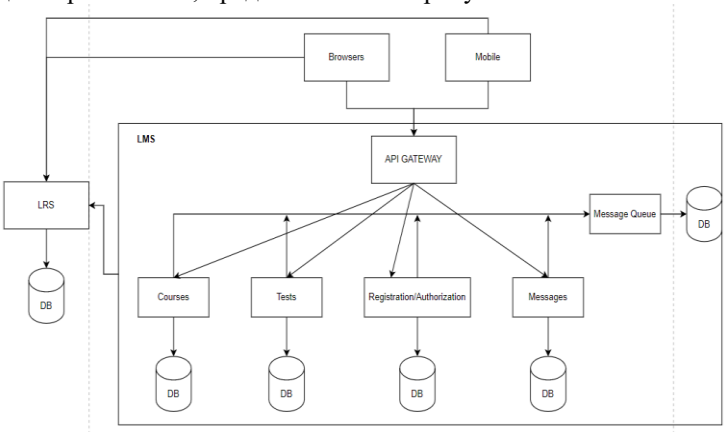


Рисунок 1 – Архитектура системы

Разберём элементы на данной схеме.

1. Browsers/Mobile – клиентские устройства, которые могут обращаться к LMS и к LRS по протоколу HTTP.

2. API GATEWAY – сервис, через который проходят все запросы с клиентской стороны. Его задача – распределение запросов на соответствующие сервисы.

3. Message Queue (очередь сообщений) – сервис, представляющий собой буфер для временного хранения сообщений. В сообщениях могут содержаться запросы, ответы, ошибки и иные данные, передаваемые между программными компонентами. За счет этого сервиса организуется связь между всеми написанными сервисами.

4. LRS – отдельная система, сохраняющая действия пользователя

5. Courses/Tests/Registration/Authorization/Messages – сервисы, определяющие основные возможности LMS.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. E-learning сегодня или какие LMS сегодня используется? — URL: https://habr.com/ru/post/94271/ (дата обращения:25.10.2022)

2. Форматы электронного обучения – URL: https://lmslist.ru/aicc-scorm-tincan-cmi5/?ysclid=lb3ttk129r992270222 (дата обращения:10.11.2022)

3. Крис Ричардсон. Микросервисы. Паттерны разработки и рефакторинга - СПб.: Питер, 2019, 544 с.: ил. — (Серия «Библиотека программиста»), ISBN 978-5-4461-0996-8

**УДК 004.94**

**Волков Е.П.**

# ИССЛЕДОВАНИЕ ПОСТРОЕНИЯ СКОРИНГОВЫХ СИСТЕМ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В данной статье рассмотрены виды автоматизированных скоринговых систем, применяемых в финансовых организациях. Спроектирован алгоритм скоринговой системы, основанный на модели, использующей множественную линейную регрессию.

При оценке физических людей современные скоринговые системы обладают двумя преимуществами – упрощением анализа факторов надёжности заёмщика и автоматизацией процесса принятия решения о выдаче кредита.

Автоматизированные системы скоринга – один из основных методов оценки риска кредитования физических лиц. Скоринг является статистико-математическим инструментом, на основе которого происходит ретроспективный анализ кредитной активности клиентов банка, что определяет вероятность надёжности (или наоборот, ненадёжности) потенциального заёмщика. Скоринговые системы в основном используются для анализа платёжеспособности физических лиц.

Цель работы – исследование и разработка скоринговой системы, основанной на множественной линейной регрессии.

Для достижения поставленной цели, были выделены следующие задачи:

1. Исследовать виды скоринговых систем.
2. Спроектировать алгоритм работы системы.

Множественная линейная регрессия соотносит зависимую переменную (платёжеспособность клиента) с линейной функцией ряда независимых переменных (скоринговыми характеристиками). Уравнение множественной линейной регрессии имеет вид: $Y=b_0+b_1x_1+\ldots+b_nx_n$, где $b_n$ вычисляется при помощи метода наименьших квадратов.

Метод наименьших квадратов (МНК) — математический подход для оценки параметров моделей на основании экспериментальных данных, содержащих случайные ошибки. Если данные известны с некоторой погрешностью, то вместо неизвестного точного значения параметра модели используется приближенное. Поэтому параметры модели должны быть рассчитаны так, чтобы минимизировать разницу между экспериментальными данными и теоретическими. Мерой рассогласования между фактическими значениями и значениями, оцененными моделью в методе наименьших квадратов, служит сумма

квадратов разностей между ними, т.е.: $\sum_{i=1}^{N}(y' - y)^2$, где y’ – оценка, полученная с помощью модели, y – фактически наблюдаемое значение.


Рисунок 1 – Алгоритм скоринговой системы

Множественная линейная регрессия находит наилучшую линейную зависимость путём построения линии регрессии (рисунок 1) с минимальной суммой среднеквадратичных отклонений от имеющихся факторов. Стоит отметить, что отклонение (остаток) отдельной точки от линии регрессии (от предсказанного значения) подчиняется закону нормального распределения Гаусса. Это даёт основание полагать, что при длительном сохранении исходных данных без изменения, процент попадания случайной величины на отрезки, равные стандартному отклонению будет существенно уменьшаться в случае отклонения от средних значений. Из этого можно сделать вывод, что при нахождении наиболее подходящего предсказывающего фактора, вероятность определения платёжеспособности или неплатёжеспособности клиента будет увеличиваться по мере сохранения исходных данных без изменения.

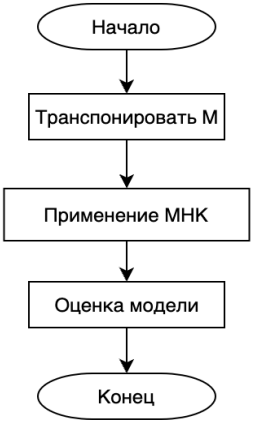


Рисунок 2 – Алгоритм процесса моделирования

В данной работе были рассмотрены понятия скоринга, скоринговых моделей, которые используются в системах. Особенности реализации модели, основанной на множественной линейной регрессии

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Беляев, М.К. Специфические риски потребительского кредитования. – М.: Элит, 2006. – 56 с.

2. Вахрушев Д.С., Риск-менеджмент в коммерческом банке: теоретические основы и проблемы организации в России. - М.: Граница, 2004. - 317 с

3. Cauoette J.B., Altman E. I., Narayanan P: Managing credit risk: The next great financial challenge. — L.: John Wiley & Sons, Inc., 1998.

**УДК 004.855.5**

**Гладышев Б. А.**

# ИССЛЕДОВАНИЕ СЕМЕЙСТВА АЛГОРИТМОВ МЕТОДА ОПОРНЫХ ВЕКТОРОВ (SVM)

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В статье представлены результаты исследования теоретической основы алгоритмов SVM. Приводится описание математической постановки задачи классификации в случае линейно разделимых данных, а также способы построения классификатора, когда данные нельзя разделить линейно.

Метод опорных векторов представляет собой группу алгоритмов машинного обучения с учителем, которые используются в задачах классификации и регрессии [1]. Идея метода заключается в построении гиперплоскости, разделяющей обучающую выборку на 2 класса в пространстве признаков. Разделяющая гиперплоскость строится так, чтобы ширина создаваемого ею зазора между категориями объектов была максимальной, поскольку в этом случае предполагается наиболее точная работа классификатора. Полученные в ходе работы метода параметры гиперплоскости в дальнейшем используются для определения категорий новых объектов в зависимости от того, по какую сторону зазора они оказываются.

Объекты в алгоритмах SVM рассматриваются как векторы (точки) в векторном пространстве признаков [2]. Каждое измерение пространства представляет собой множество значений соответствующего признака. Сами значения являются действительными числами, поэтому в случае данных, представленных в другом виде, необходимо преобразовать их в числовую форму каким-либо способом.

Пусть далее X – множество векторов пространства Rn, являющихся описаниями классифицируемых объектов, а Y = {-1, 1} – метки двух классов, к которым однозначно могут быть отнесены объекты из X. Имеется обучающая выборка – подмножество объектов из X, для которых известна их классовая принадлежность. Алгоритм классификации представляет собой отображение, которое сопоставляет каждый объект с одним из двух классов:

$$f(\boldsymbol{x}): X \to Y, \quad (1)$$

Это отображение также называют функцией классификации. Она принимает значение 1 для объектов первого класса и -1 для объектов второго. Здесь и далее жирное выделение переменной является обозначением вектора. Метод опорных векторов в базовом виде является алгоритмом бинарной классификации и выполняет её, основываясь на предположении, что наилучший результат будет достигнут при оптимальном разделении гиперплоскостью двух классов между собой в пространстве. Оптимальным является такое разделение, при котором достигается максимальная ширина зазора между классами. Его можно изобразить в виде пространства между двумя гиперплоскостями, параллельными разделяющей, при этом каждая гиперплоскость-граница включает в себя один или несколько векторов ближайшего класса. Оптимальная разделяющая гиперплоскость равноудалена от границ зазора.

Пример, демонстрирующий оптимальную разделяющую гиперплоскость в двумерном пространстве, представлен на рисунке 1. Объекты первого класса изображены квадратами, а второго – кругами.

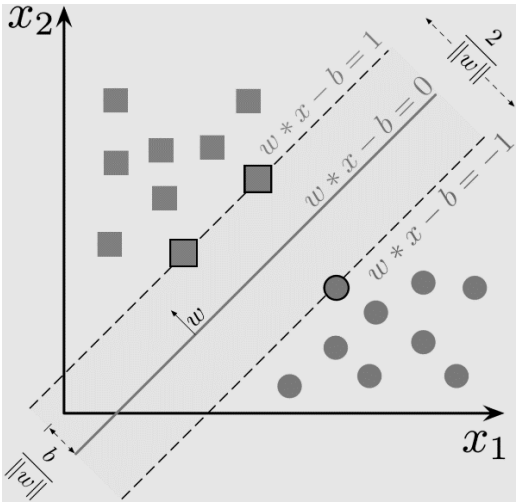


Рисунок 3 – Оптимальная гиперплоскость в пространстве $R^2$

Разделяющая гиперплоскость нарисована сплошной линией, границы зазора показаны пунктиром. В пространстве $R^n$ уравнение гиперплоскости определяется формулой

$$\langle \boldsymbol{w},\boldsymbol{x} \rangle + b = 0, \tag{2}$$

Угловые скобки – скалярное произведение векторов (на рисунке записано через звёздочку). $\boldsymbol{w}$ – направляющий вектор гиперплоскости, $b$ – её смещение относительно нулевой точки. Обведённые объекты, лежащие на границах зазора – это опорные векторы, в честь которых метод и получил своё название. Они называются опорными, так как именно они определяют зазор между классами и, следовательно, параметры разделяющей гиперплоскости. Процесс обучения алгоритма заключается в решении задачи максимизации зазора между классами, который выражается величиной, обратно пропорциональной норме направляющего вектора $\|\boldsymbol{w}\|$. При этом должно получиться так, чтобы объекты обучающей выборки, принадлежащие разным классам, оказались по разные стороны гиперплоскости. Таким образом, ставится задача нелинейной оптимизации вида

$$\frac{\|\boldsymbol{w}\|^2}{2} \to \min_{\boldsymbol{w},b}$$

при ограничениях

$$\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b \geq 1, \forall \boldsymbol{x} \in C_1$$
$$\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b \leq -1, \forall \boldsymbol{x} \in C_2$$

или

$$y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b) \geq 1 \tag{3}$$

Для опорных векторов значение левой части ограничения будет совпадать с числом в правой. Множитель $y_i$ в нижней записи – метка класса для вектора $\boldsymbol{x}_i$ обучающей выборки. В случае линейно разделимых данных такая задача оптимизации имеет единственное решение, а функцию классификации при этом можно записать в виде

$$f(x) = \operatorname{sgn}(\langle \boldsymbol{w}^*, \boldsymbol{x} \rangle + b^*), \tag{4}$$

где $\boldsymbol{w}^*$ и $b^*$ – параметры оптимальной гиперплоскости.

Если данные не являются линейно разделимыми, то при построении классификатора прибегают к нескольким способам ухода от неразделимости.

Во-первых, это фильтрация обучающей выборки, которая направлена на отсев выбросов в данных, то есть, одиночных векторов, выбивающихся из основной области своего класса, которые попадают слишком близко к зазору или по другую сторону него в область другого класса.

Также существует подход, называемый классификацией с мягким зазором, при котором вводят набор дополнительных переменных $\varepsilon_1, \ldots, \varepsilon_l$ (их количество в общем случае равно размеру обучающей выборки). Они показывают, насколько вектор нарушает условие разбиения гиперплоскостью. Эти переменные добавляются в неравенства ограничений вместе с соответствующим вектором обучающей выборки, а их сумма с калибровочными коэффициентами вводится в целевую функцию. Таким образом, задача поиска параметров разделяющей гиперплоскости принимает вид

$$\frac{\|\boldsymbol{w}\|^2}{2} + C\sum_{i=1}^{l} (\varepsilon_i)^k \to \min_{\boldsymbol{w}, b, \varepsilon}$$

при ограничениях , (5)

$$y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b) \geq 1 - \varepsilon_i$$

$$\varepsilon_i \geq 0$$

Переменная $C$ в целевой функции служит коэффициентом, позволяющим балансировать между максимизацией зазора и минимизацией количества ошибок при нахождении параметров на обучающей выборке [2]. Показатель $k$ позволяет влиять на величину отклонения от случая линейной разделимости, усиливая влияние векторов, расположенных более близко к разделяющей гиперплоскости. По сути, цель использования метода опорных векторов с мягким зазором – это разрешение классификатору ошибаться при обучении на линейно-неразделимой выборке с целью получения параметров разделяющей

гиперплоскости, которая могла бы достаточно точно работать на основном объёме данных. Стоит отметить, что опорные векторы каждого из классов в этом случае уже не будут лежать по разные стороны от гиперплоскости, как это было бы в случае линейной разделимости.

Ещё один способ, с помощью которого решают проблему линейно-неразделимых данных, это применение ядер в методе опорных векторов [1]. Применительно к задаче классификации ядро – это функция вида

$$K(\boldsymbol{x}_1, \boldsymbol{x}_2): X \times X \to \mathrm{R}, \tag{6}$$

если она представима в виде

$$K(\boldsymbol{x}_1, \boldsymbol{x}_2) = \langle \psi(\boldsymbol{x}_1), \psi(\boldsymbol{x}_2) \rangle, \tag{7}$$

при некотором отображении

$$\psi : X \to H, \tag{8}$$

где $H$ – пространство со скалярным произведением, называемое также спрямляющим пространством [4].

То есть, ядро – это отображение пары элементов множества $X$ в множество действительных чисел. В стандартном SVM в качестве ядра применяется скалярное произведение, но в случае линейно-неразделимых данных его можно заменить другой трансформацией. Обычно спрямляющее пространство не строят явно, а вместо подбора отображения $\psi$ пользуются хорошо известными функциями ядра, подходящими для большинства задач классификации. Например, вид одного из часто применяемых ядер – радиальной базисной функции:

$$K(\boldsymbol{x}_1, \boldsymbol{x}_2) = e^{-\gamma \|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2}, \gamma > 0.$$

Целью применения ядер является преобразование линейно-неразделимых данных из $n$-мерного пространства в более высокую размерность, в которой они становятся линейно-разделимыми. Пример, демонстрирующий переход из $\mathrm{R}^2$ в $\mathrm{R}^3$ представлен на рисунке 2 [3].

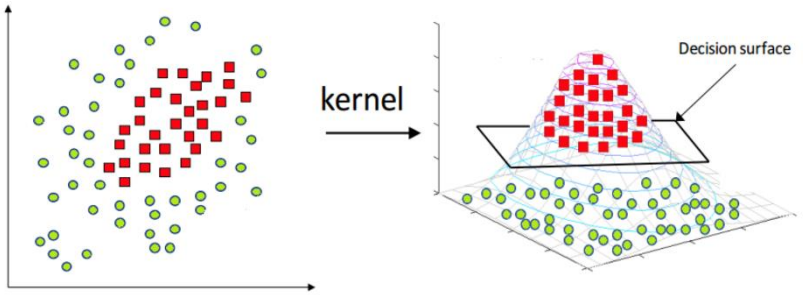


Рисунок 4 – Наглядная демонстрация применения ядер в SVM

При применении ядер в методе опорных векторов в целевой функции задачи поиска параметров оптимальной разделяющей

гиперплоскости и, соответственно, функции классификации скалярные произведения векторов заменяются на использованное ядро-отображение [3].

На практике при реализации SVM задачу нахождения параметров разделяющей гиперплоскости (1) решают в форме двойственной задачи Лагранжа, поскольку это позволяет преобразовать вычислительно сложные неравенства системы ограничений со скалярным произведением векторов в более простую форму равенств, которые проще вычислять и соблюдать в поиске решения [2]. Переход от прямой задачи (1) к двойственной задаче Лагранжа можно выполнить следующим образом. Двойственная функция определяется через Лагранжиан

$$L(\boldsymbol{w},b,\boldsymbol{\alpha}) \to \min_{\boldsymbol{w},b}, \tag{9}$$

который записывается в виде [2]

$$L(\boldsymbol{w},b,\boldsymbol{\alpha}) \to \frac{1}{2}\sum_{k=1}^{n} w_k^2 - \sum_{i=1}^{l}\alpha_i\left(y_i\left(\sum_{k=1}^{n} w_k x_{ik} + b\right) - 1\right), \tag{10}$$

Вектор $\boldsymbol{\alpha}$ представляет собой множители Лагранжа. $n$ – размерность пространства, а $l$ – размер обучающей выборки. Поскольку Лагранжиан здесь представляет собой выпуклую функцию, для любого вектора $\boldsymbol{\alpha}$ глобальный минимум $L$ будет достигаться только в случае равенства нулю его градиента, то есть, задача может быть переписана в форме

$$L(\boldsymbol{w},b,\boldsymbol{\alpha}) \to \max_{\boldsymbol{w},b,\boldsymbol{\alpha}}$$

$$\text{при} \quad , \tag{11}$$

$$\nabla_{\boldsymbol{w},b} L(\boldsymbol{w},b,\boldsymbol{\alpha}) = \mathbf{0}$$

**0** представляет собой нулевой вектор. Из этого ограничения можно вывести следующее [2]

$$w_k = \sum_{i=1}^{l}\alpha_i y_i x_{ik}, k = 1,2,\ldots.n$$

$$, \tag{12}$$

$$\sum_{i=1}^{l}\alpha_i y_i = 0$$

После подстановки в Лагранжиан этих уравнений задача преобразуется к виду [2]

$$\sum_{i=1}^{l}\alpha_i - \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j y_i y_j \left\langle \boldsymbol{x}_i, \boldsymbol{x}_j \right\rangle \to \max_{\alpha}$$

при ,(13)

$$\sum_{i=1}^{l}\alpha_i y_i = 0$$

$$\alpha_i \geq 0, \quad i = 1,2,\ldots,l$$

Именно такую форму задачи поиска параметров оптимальной разделяющей гиперплоскости автор этой статьи использовал для реализации бинарного классификатора при написании научно-исследовательской работы.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Вьюгин В.В. «Математические основы машинного обучения и прогнозирования» М.: 2013, 2018 - 484 с.

2. Alexey Nefedov. Support Vector Machines: A Simple Tutorial.— 2016.

3. Rizwan A, Iqbal N, Ahmad R, Kim D-H. WR-SVM Model Based on the Margin Radius Approach for Solving the Minimum Enclosing Ball Problem in Support Vector Machine Classification. [Электронный ресурс]. 2021;11(10):4657. URL: http://dx.doi.org/10.3390/app11104657

4. К. В. Воронцов Математические методы обучения по прецедентам. [Электронный ресурс].

URL: http://www.machinelearning.ru/wiki/images/6/6d/Voron-ML-1.pdf

**УДК 005.519.8**

**Головкин Н.В., Цуканова Н.И.**

# ВЛИЯНИЕ МЕТОДОВ НОРМАЛИЗАЦИИ СЛОВ НА ТОЧНОСТЬ КЛАССИФИКАЦИИ ТЕКСТА

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

Рассматриваются различные методы нормализации слов и их влияние на точность классификации текста с помощью различных типов нейронных сетей

Важным шагом при обучении нейронной сети для решения задачи классификации текста является предобработка текста. Один из этапов предобработки текста - это нормализация слов, т.е. приведение их к единому виду.

В данной статье рассматриваются такие методы нормализации слов, как стемминг и лемматизация.

Стемминг – это нахождение основы слова. Основа слова не всегда совпадает с его корнем. Сам термин стемминг (stemming) образован от слова «stem» – ствол, стебель, основа.

Например, при стемминге слов «виноградная», «бежал» и «красиво» они примут вид «виноградн», «бежа», «красив».

Лемматизация – это приведения словоформы к лемме – её словарной форме. В русском языке словарными формами считаются следующие морфологические формы:

для существительных – именительный падеж, единственное число;

для прилагательных – именительный падеж, единственное число, мужской род;

для глаголов, причастий, деепричастий – глагол в неопределённой форме несовершенного вида.

Например, при лемматизации слов «виноградная», «бежал» и «красиво» они примут вид «виноградный», «бежать», «красиво».

Для исследования влияния методов нормализации слов на точность классификации текста с помощью нейронных сетей были выбраны полносвязная нейронная сеть и нейронная сеть LSTM.

Полносвязная сеть:

```
model = Sequential()
model.add(Dense(512,
activation='relu', input_dim=X_train.shape[1]))
model.add(Dropout(0.81))
model.add(Dense(512, activation='relu'))
model.add(Dropout(0.81))
model.add(Dense(11, activation='softmax'))
```

Нейронная сеть LSTM:

```
model = Sequential()
model = Sequential()
model.add(Embedding(1000,100,
input_length=X_train.shape[1]))
model.add(SpatialDropout1D(0.2))
model.add(LSTM(100, dropout=0.2))
model.add(Dense(11, activation='softmax'))
```

Кроме использования методов нормализации слов примеры, подаваемые на вход нейронной сети, также были предобработаны следующим образом:

удалены числа;

удалены ведущие и конечные пробелы;

удалены знаки препинания;

все символы приведены к нижнему регистру;

примеры разбиты на токены;

удалены стоп-слова.

Для тестирования нейронных сетей использовался набор данных на русском языке, состоящий из 11 классов. В большей части из них находится по 5000 примеров, в других – приблизительно 1000. Размер словаря – 5000.

Стемминг осуществлялся с помощью библиотеки NLTK, а лемматизация – PyMorphy2.

Результаты тестирования приведены ниже в таблице 1.

Таблица 1 – Результаты исследования

| Нейронная сеть | Метод нормализации слов | Точность на тестовой выборке |
|---|---|---|
| Полносвязная | - (токенизация без предобработки) | 95,9% |
| Полносвязная | - (токенизация с предобработкой) | 95,4% |
| Полносвязная | Стемминг | 95,5% |
| Полносвязная | Лемматизация | 95,9% |
| LSTM | - (токенизация без предобработки) | 93,3% |
| LSTM | - (токенизация с предобработкой) | 93,6% |
| LSTM | Стемминг | 94,8% |
| LSTM | Лемматизация | 95,3% |

Исходя из результатов можно сделать вывод, что метод нормализации слов не оказывает значимого влияния на точность при использовании полносвязной сети. Влияние механизма нормализации слов проявляется при использовании нейронной сети LSTM, а именно, при использовании слоя Embedding. Использование лемматизации позволило повысить точность классификации на 2% по сравнению с токенизацией без предобработки.

Стоит отметить, что на точность влияют и гиперпараметры нейронной сети, поэтому нельзя с полной уверенностью сказать, что 2% – это максимально достижимый прирост от использования предобработки текста.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Лейн Х. Обработка естественного языка в действии / Лейн Х., Хапке Х., Ховард К. — СПб.: Питер, 2020. ─ 576 с.

2. Ганегедара Т. Обработка естественного языка с TensorFlow / пер. с англ. ─ М.: ДМК-Пресс, 2020. ─ 382 с.

**УДК 004.056.5**

**Кибамба Ж.Ж., Филатов И.Ю.**

# РЕАЛИЗАЦИЯ АЛГОРИТМА ОБРАБОТКИ И ОПТИМИЗАЦИИ ТРАНЗАКЦИИ В СИСТЕМЕ ЧАСТНЫХ ДЕНЕЖНЫХ ПЕРЕВОДОВ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

Целью данной работы является описание и реализация алгоритма обработки и оптимизации транзакции денежных переводов, осуществляемых физическими лицами.

Частные денежные переводы [1] – денежные переводы, которые осуществляются как частной группой, состоящей из четырех человек (минимум). Одна пара в каждом конечном пункте, состоящая из одного

человека, которому нужно отправить деньги в второй пункт, и другого, которому нужно их получить от второго пункта (рисунок 1).

SA1 и SB1 – отправителя в каждом пункте, а RA1 и RA2 – получателя. Так как отправитель SA1 не может отправить деньги получателю RB1 из-за нынешней ситуации, такая проблема тоже для отправителя SB1 и получателю RA1, то необходимо реализовать систему частных денежных переводов (рисунок 2).

Таким образом, отправитель SA1, находящийся в пункте А отправляет денежную сумму A1 получателю RA1, который должен был получить её из пункта Б от отправителя SB1. И то же самое в пункте Б отправитель SB1, который должен был отправить денежную сумму A2 получателю RA1, отправляет эту получателю RB1, который должен был получить её от отправителя SA1 из пункта А.

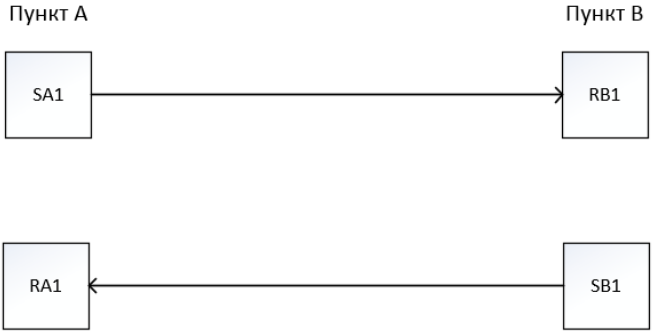


Рисунок 1 – Обычная схема p2p денежного перевода

Эта логика не пойдет если количество людей увеличивается, если не только одна пара людей в каждом конечном пункте, а более двух человек в пункте А и в пункте Б (рисунок.3), то необходимо оптимально распределить разные суммы между всеми частными лицами. Для этого будем применить теорию, которая похожа на теорию транспортной задачи [2].

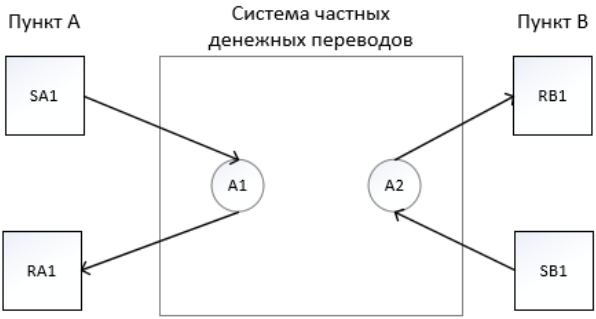


Рисунок 2 – Общая схема p2p системы частных денежных переводов

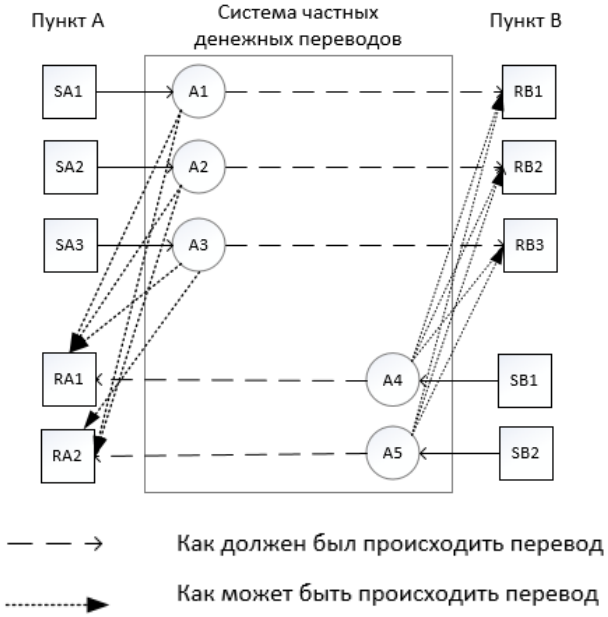


Рисунок 3 – Общая схема p2p системы частных денежных переводов с несколькими участниками

Рассмотрим данные из одного пункта, пусть из пункта А (таблица 1).

Таблица 1 – Данные пункта А

| | **Приемники** | |
|---|---|---|
| **Отправители** | *RA1* | *RA2* |
| *SA1* | D14 | D15 |
| *SA2* | D24 | D25 |
| *SA3* | D34 | D35 |

В таблице 1 написаны данные из пункта А в виде таблицы транспортной задачи, где в качестве поставщики – отправители, а потребители – приемники. Dij – стоимость перевозки – абсолютное значение разницы между суммой денег отправителя i и суммой денег, которую должен получить приемник j. Здесь же речь не идет об составе плана перевозок, позволяющий полностью вывести продукты (суммы денег) всех отправителей, полностью обеспечивающий потребности всех приемников.

Нам надо только оптимально распределить денежные переводы между всеми частными лицами. То есть если участников несколько, пусть будет достаточно сбалансированный обмен, чтобы один не отправлял все свои деньги, а другой ничего не отправлял. Для реализации этой задачи посмотрим общий алгоритм и алгоритм процесса обработки транзакции:

Шаг 1. Выбирается одно из двух конечных пунктов, в котором сумма денег отправителей больше, чем сумма денег приемников. Если сумма одинакова в обоих пунктах, то выбирается любой из них.

Шаг 2. Вычисляются разницы $D_{ij}$ для каждого отправителя и приемника, где

$$D_{ij} = \left| A_i - A_j \right|$$

Шаг 3. Удаляются отправитель и получатель у которых значение разницы $D_{ij}$ равно нулю из списка участников обработок. Тогда для этих удаляющих участников отправитель отправляет все свои деньги получателю напрямую без каких-либо обработок.

Шаг 4. Идет процесс обработки транзакции если список обработок не пустой (рисунки 4 и 5). По очереди каждый отправитель отправляет сумму $D_{ij}$ одному приемнику если это возможно, далее операция повторяется для следующего приемника и так далее. Каждый раз после этой операции сумму денег обоих участников транзакции уменьшается на $D_{ij}$. В конце цикла проверяется если есть возможность продолжать, то есть если в системе существует, ходя один отправитель, у которого остались деньги и один приемник, который не получил всю сумму, которая ему была надо.

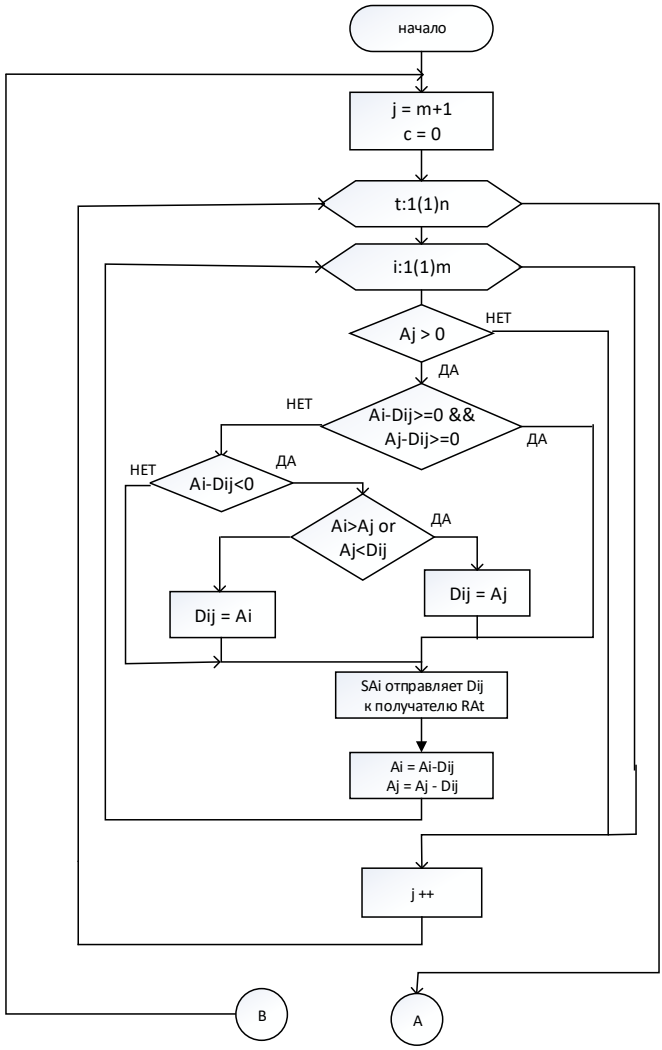


Рисунок 4 – Алгоритм процесса обработки транзакции (начало)

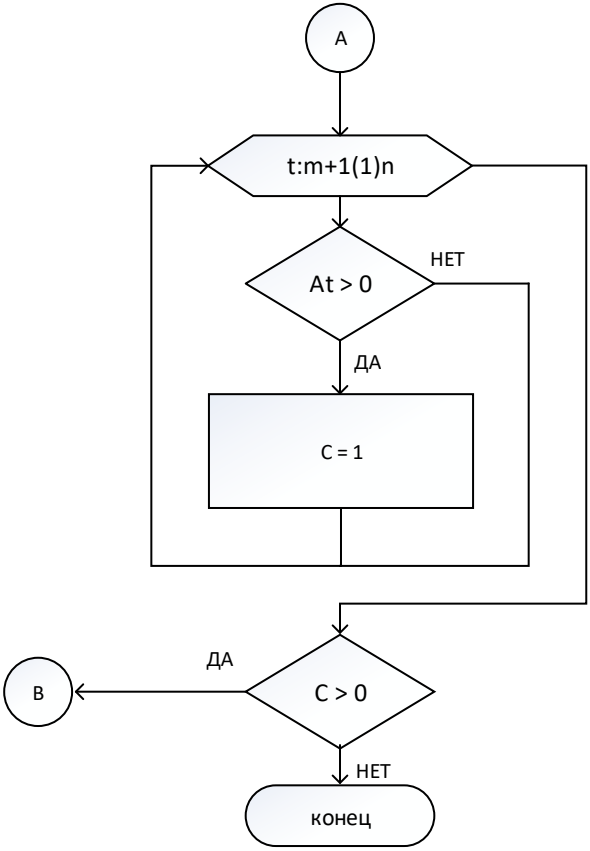


Рисунок 5 – Алгоритм процесса обработки транзакции (конец)

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Кибамба Ж.Ж., Исследование и реализация методов автоматизации частных денежных переводов [Текст] // Научная исследовательская работа – НИР - 2022. - С.9.

2. Бунтова Е.В., Использование транспортной задачи для определения оптимального плана грузоперевозок. // Е.В. Бунтова, – Самара, 2014. – С.5-9

**УДК 004.021**

**Климкин К.С., Цуканова Н.И.**

# РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ОБРАБОТКИ ИЗОБРАЖЕНИЙ ОТПЕЧАТКОВ ПАЛЬЦЕВ С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА ДАННЫХ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В статье рассмотрены способы улучшения качества сканирующих устройств благодаря дополнительному тестированию с помощью зашумленных изображений, а также благодаря устранению шума из входных данных.

В настоящее время обработка изображений отпечатков пальцев используется повсеместно: будь то криминалистика или сканер отпечатков пальцев. Для улучшения качества сканирующих устройств необходимо тщательное их тестирование, которое предполагает выявление и устранение наиболее значимых уязвимостей.

Одним из таких способов улучшения является проверка с зашумленными изображениями. Это даёт возможность выявить наиболее уязвимые места, так как нельзя быть полностью уверенным в качестве входных данных. В python это можно сделать с помощью функции numpy.random.normal(loc=0.0, scale=1.0, size=(размер исходного изображения)) [1], которая заполняет матрицу значениями нормального распределения. Эта матрица умножается на коэффициент шума, а после складывается с матрицей исходного изображения. В результате получаем изображения с шумом (рисунок 1). Управляя этим коэффициентом можно создавать отпечатки пальцев с разной степенью зашумления, что позволит выявить уровень, при котором теряется способность распознавания у сканирующий устройств.

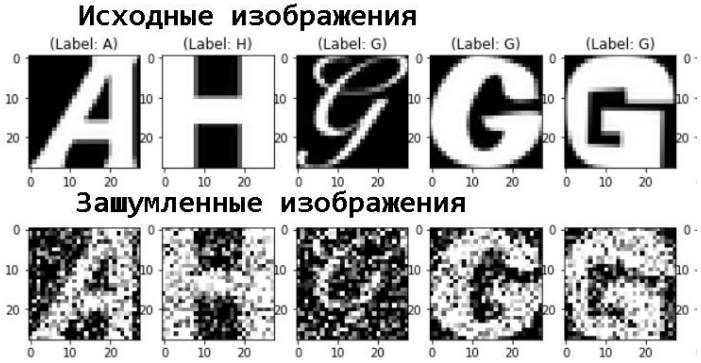


Рисунок 1 – Пример работы функции python numpy.random.normal

Ещё один способ улучшения сканирующих устройств - это устранение избыточного шума изображения. Для этого можно использовать шумоподавляющий автокодировщик [1]. Он состоит из кодера, скрытого слоя и декодера (рисунок 2).

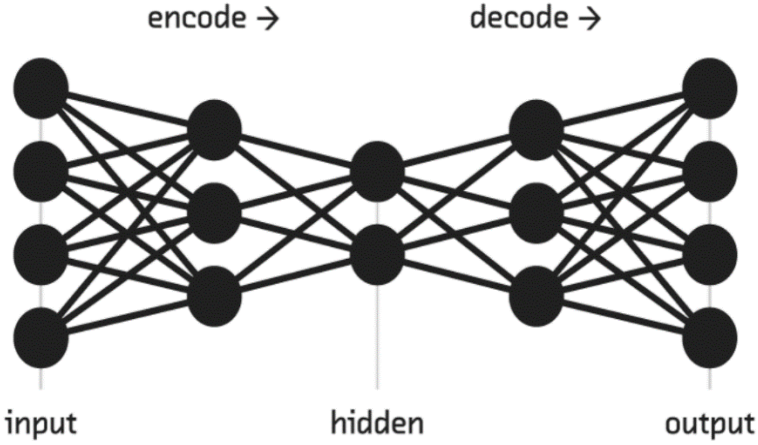


Рисунок 2 – Архитектура автокодировщика

Суть автокодировщика состоит в том, что на вход подаются зашумлённые изображения, а на выходе нужно получить изображения без шума, наиболее приближенные к исходным. Для зашумления можно воспользоваться все той же функцией python numpy.random.normal. Подготовленные данные подаются на автокодировщик. Кодер сжимает входные данные в скрытое пространство, а затем декодер восстанавливает данные в соответствии с этим пространством для получения окончательных выходных данных. На каждом шаге обучения сравнивается восстановленное изображение с исходным. Обучение длится до тех пор, пока разница между ними не станет минимальной. В результате получаются восстановленные изображения без шумов. Получение более четких изображений позволит повысить точность сканирующих устройств. А в связке с тестированием с зашумленными изображениями позволит найти наилучшие пороговые значения качества сканирующего устройства, дальше которых нет смысла улучшать способность к распознанию отпечатков пальцев.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1.Пример модели глубокого обучения в keras: Шумоподавляющий автокодировщик [Internet].[cited 2022 December 25]. Available from: https://pythonru.com/biblioteki/glubokoe-obuchenie-v-keras

**УДК 004.021**

**Кондрашов Д.И., Филатов И.Ю.**

# АНАЛИЗ ЭФФЕКТИВНОСТИ АЛГОРИТМА MAPREDUCE ПРИ РЕШЕНИИ ЗАДАЧИ ПОДСЧЁТА КОЛИЧЕСТВА ВХОЖДЕНИЙ РАЗЛИЧНЫХ СЛОВ В ТЕКСТ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В статье рассматривается эффективность работы алгоритма по обработке больших данных MapReduce в сравнении с линейным алгоритмом. Оба выполняют задачу по подсчёту количества вхождений различных слов в текст. Алгоритмы реализованы на языке программирования Java.

MapReduce – технология, созданная Google, для выполнения параллельных вычислений над очень большими, вплоть до нескольких петабайт, наборами данных в компьютерных кластерах [3].

Название происходит от двух функций высшего порядка (функция, принимающая в качестве аргументов другие функции или возвращающая другую функцию в качестве результата): map (применяет какую-либо функцию к каждому элементу списка своих аргументов, выдавая список результатов как возвращаемое значение) и reduce (производит преобразование структуры данных к единственному атомарному значению при помощи заданной функции) [3].

Алгоритм MapReduce состоит из двух шагов: Map и Reduce. На Map - шаге происходит предварительная обработка входных данных. На Reduce - шаге происходит свёртка предварительно обработанных данных.

Каноническим примером применения технологии MapReduce является решение задачи подсчёта количества вхождений различных слов в текст. На шаге Map весь текст разбивается на кортежи вида «слово, 1», на шаге Reduce единички с одинаковым словом складываются. Пример: текст «foo bar baz bar». Результат шага Map:

[ 'foo', 1 ]
[ 'bar', 1 ]
[ 'baz', 1 ]
[ 'bar', 1 ]

Результат шага Reduce:

[ 'bar', 2 ]
[ 'baz', 1 ]
[ 'foo', 1 ]

Преимущество алгоритма MapReduce заключается в том, что он позволяет распределено производить операции предварительной обработки и свертки.

Упрощённая схема работы алгоритма MapReduce приведена на рисунке 1.

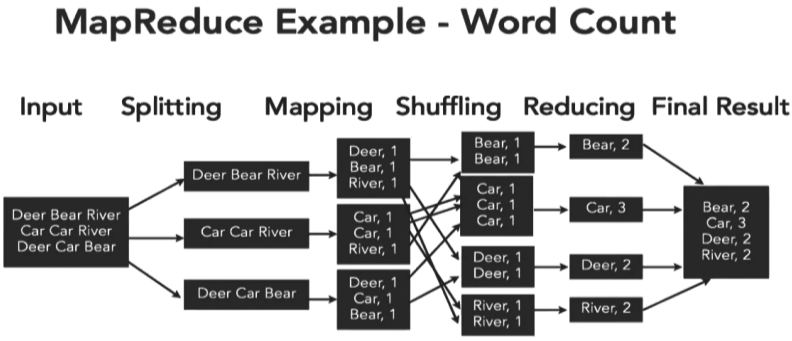


Рисунок 1 — Упрощённая схема работы алгоритма MapReduce

Попытки сравнить эффективность алгоритма MapReduce уже проводились, например, магистрантом Диденко А.А. в 2018 году[2], где выполнялось сравнение с линейным алгоритмом, но для решения специфической задачи подсчёта суммарной выручки компаний. Данные агрегировались из транзакционного файла.

Учитывая выводы работы Диденко, для достижения максимальной объективности, при сравнении работы алгоритмов, авторы настоящей статьи взяли за основу каноническую задачу по подсчёту количества вхождений различных слов в текст, используя при этом, другой язык реализации алгоритмов.

Для проверки эффективности работы алгоритма MapReduce в рамках решения выше описанной задачи, была произведена его реализация на языке Java. При этом использовались библиотеки «org.apache.hadoop», в том числе «org.apache.hadoop.mapreduce». Компания Apache упростила работу с большими данными за счёт реализации большинства алгоритмов их обработки, в том числе и MapReduce, в своих библиотеках. Подробная информация по использованию специализированных средств разработки и тематических библиотек приведена на официальном сайте Apache [1].

Программный код метода, реализующего линейный алгоритм, с использованием Stream API, написанный на языке Java, решающий задачу по подсчёту количества вхождений различных слов в текст, приведён на рисунке 2.

```
private static final String RELATIVE_PATH_TO_FILE = "src\\main\\resources\\fileOut";
public static void countWordsInTextFile(String filePath) throws IOException {
    FileWriter fileWriter = new FileWriter(RELATIVE_PATH_TO_FILE);
    PrintWriter printWriter = new PrintWriter(fileWriter);
    try (Stream<String> lines = Files.lines(Paths.get(filePath))) {
        lines.flatMap(line -> Arrays.stream(line.trim().split( regex: "\\s+"))) Stream<String>
                .map(word -> word.replaceAll( regex: "\\P{L}", replacement: "").trim())
                .map(String::toLowerCase)
                .filter(word -> word.length() > 0)
                .collect(Collectors.groupingBy(Function.identity(), Collectors.counting())) Map<String, Long>
                .entrySet().forEach(printWriter::println);
    } catch (IOException e) {
        e.printStackTrace();
    }
}
```

Рисунок 2 — Код метода, выполняющего подсчёт количества вхождений различных слов в текст, линейный алгоритм

Принцип работы метода следующий. Текст для подсчёта слов загружается из входного файла (переменная «filePath») и результат помещается в выходной файл (константа «RELATIVE_PATH_TO_FILE»). Исходный текст разделяется на отдельные слова с помощью регулярного выражения, из слов убираются все не языковые символы (пробелы, точки, запятые и т.д.), все слова приводятся к нижнему регистру, для тех слов, у которых длина больше нуля выполняется их группировка в карту («Map<String, Long>») с подсчётом количества вхождений каждого из них, далее все записи вида «ключ-значение» из карты записываются в выходной файл.

В качестве входного файла был использован вручную сгенерированный текстовый файл (с расширением «.txt») с набором слов через запятую. Один и тот же файл подавался на вход двух разных программ. Размер файла составляет 172 856 КБ.

Сравнение работы алгоритмов проводилось на базе ноутбука «Dell Vostro 15 3515». Версия JDK: «Oracle OpenJDK version 18.0.1». Интегрированная среда разработки: «IntelliJ IDEA 2022.2.4 (Community Edition)». Процессор: «AMD Ryzen 7 3700U with Radeon Vega Mobile Gfx 2.30 GHz». Размер оперативной памяти: 16,0 ГБ (доступно: 13,9 ГБ). Операционная система – 64-разрядная, «Windows 10 Pro», версия «21H2», сборка «19044.2130».

При таких условиях, в результате исследования выяснилось, что для решения задачи по подсчёту количества вхождений различных слов в текст с одинаковыми исходными данными, алгоритм MapReduce, за счёт параллельного выполнения задач, оказался эффективнее линейного алгоритма в 32 раза. Время выполнения алгоритма MapReduce составило 6,87 секунд. Время выполнения линейного алгоритма составило 219,85 секунд.

Таким образом, в данной статье была рассмотрена эффективность алгоритма MapReduce в сравнении с линейным алгоритмом. Использовались: язык программирования Java, библиотеки Apache

Hadoop - для реализации алгоритма MapReduce. Итоговый результат – MapReduce эффективнее в 32 раза, что приблизительно совпадает с результатами Диденко [2].

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. MapReduce Tutorial [сайт]. – URL: https://hadoop.apache.org/docs/current/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapReduceTutorial.html (дата обращения: 20.11.2022).

2. Анализ эффективности алгоритма mapreduce при решении задачи вычисления суммарной выручки предприятий региона с помощью apache hadoop. [сайт]. URL: https://science.kuzstu.ru/wp-content/Events/Conference/RM/2018/RM18/pages/Articles/31518-.pdf (дата обращения: 27.11.2022).

3. MapReduce [сайт]. – URL: https://ru.wikipedia.org/wiki/MapReduce (дата обращения: 19.11.2022).

**УДК 004.021**

**Кошелева В.Б.**

# АНАЛИЗ АЛГОРИТМА ФОРМИРОВАНИЯ СПРАВОК В СИСТЕМЕ 1С:УНИВЕРСИТЕТ ПРОФ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

Предмет исследования – алгоритм формирования различных справок по заявкам студентов.
Цель работы – сократить временные затраты на подготовку документов и повысить эффективность работы деканата.

Деятельность университета, всех его структурных подразделений направлена на качественное и оперативное обслуживание студентов. Деканат выполняет основные и важнейшие функции университета, обеспечивая формирование различной документации, такой, например, как выдача справок для студентов.

Существует различное множество информационных систем (ИС) для автоматизации управленческой деятельности в учреждениях высшего профессионального образования. Одной из таких ИС является «1С:Университет ПРОФ». Этот программный продукт позволяет использовать внешние объекты, такие как обработки и отчеты, включающие в себя механизм системы компоновки данных (СКД). Обе функциональности обеспечивают сокращение временных затрат на подготовку документов и повышают эффективность работы деканата.

Преимущество внешних обработок состоит в том, что новые объекты не требуется каждый раз включать в состав конфигурации и, соответственно, обновлять при этом всю информационную базу 1С.

Вместо этого можно использовать любые необходимые отчёты и обработки без внесения изменений в метаданные
базы 1С.

Обработка 1С – это объект конфигурации, который служит для изменения и преобразования данных в информационной базе 1С. Отчёт 1С – это объект конфигурации, который формирует вывод данных в удобном для восприятия пользователем виде.

Оба сохраняются в отдельные файлы и имеют следующие расширения:

- внешний отчёт – *.erf;
- внешняя обработка –*.epf.

Система компоновки данных (СКД) представляет собой механизм, основанный на декларативном описании отчётов, который содержит произвольный набор таблиц и диаграмм. Он предназначен для построения отчётов, а также вывода информации, имеющей сложную структуру [1].

Для формирования справок для студентов необходимы их персональные данные, которые находятся в отдельных справочниках.

Справочники – это прикладные объекты конфигурации 1С, предназначенные для хранения в информационной базе данных, имеющих одинаковую структуру и списочный характер [2]. В них отражена полная информация о студентах: студенческий билет, ФИО, факультет, направление и др. Также для определённых видов справок необходимы специальные данные о дисциплинах, количестве часов, наименовании кафедр и др. В документах зафиксирована успеваемость и задолженности студентов.

При заявке на получение одного из видов справок запрашивается информация о студентах и выводится либо в виде списка в табличной части документа, либо в виде отчёта.

Для первого способа необходимо, чтобы реквизиты заполнялись автоматически из справочника, а также предусмотреть возможность создания новых данных. Для этого необходимо заранее прописать функции и процедуры в модуле формы документа. Преимущество данного подхода состоит в том, что это сокращает время и количество ошибок, связанных с человеческим фактором.

Во втором способе механизм вывода уже встроен в систему и это значительно облегчает работу. Однако текст запроса должен быть полным, и при назначении параметров важно правильно их указать. В противном случае, запрос необходимо написать заново.

Общий алгоритм формирования справок представлен на рисунке 1.

Вузы используют разные информационные системы при ведении деятельности, поэтому должна быть возможность переноса данных из старой системы в систему 1С:Университет ПРОФ.

После формирования справки группировки студентов с помощью компоновки данных выводится общий список студентов. Также есть возможность сформировать отчёт по заявке на получение справок и общие сведения по конкретному обучающемуся.

Таким образом, был проанализирован алгоритм формирования справок. Можно сделать вывод о том, что в данном алгоритме целесообразно использовать внешнюю обработку данных и систему компоновку данных вместо табличной части документа.

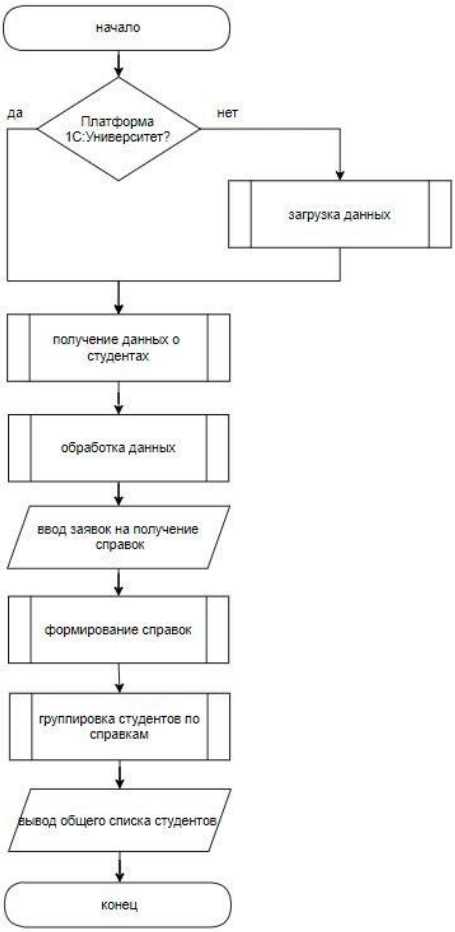


Рисунок 1 – Схема алгоритма формирования справок

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Хрусталева Е. Ю. Разработка сложных отчётов в «1С:Предприятии 8». Система компоновки данных / Е. Ю. Хрусталева. - 2-е изд. – М.: 1С-Паблишинг, 2012. – 484 с.: ил.

2. Радченко М.Г. 1С:Предприятие 8.1. Практическое пособие разработчика. Примеры и типовые приемы. – М.: 1С-Паблишинг; СПб.: Питер, 2007. – 512 с.


**УДК 519.857**

**Крашенинникова Д.А.**

# РАЗРАБОТКА АЛГОРИТМА ВОЛНОВОЙ ТРАССИРОВКИ ДЛЯ РЕШЕНИЯ ЗАДАЧИ НАХОЖДЕНИЯ КРАТЧАЙШЕГО ПУТИ МЕЖДУ ИГРОВЫМИ ПЕРСОНАЖАМИ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

Важной и актуальной задачей при проектировании игрового продукта является планирование движения игровых персонажей в условиях неопределённости, а также при наличии непреодолимых для них зон игрового поля или границ между зонами. Цель настоящей работы – разработка алгоритма волновой трассировки для решения задачи нахождения кратчайшего пути между игровыми персонажами.

На протяжении десятилетий развитие компьютерных игр тесно связано с научными открытиями мира вычислительных технологий. Множество задач при разработке компьютерных игр были решены с помощью давно известных алгоритмов динамического программирования. Важной и актуальной задачей при проектировании игрового продукта является планирование движения игровых персонажей в условиях неопределённости, а также при наличии непреодолимых для них зон игрового поля или границ между зонами.

Существует множество алгоритмов поиска кратчайшего пути, в данной работе исследуется применение алгоритма волновой трассировки.

Теоретическая часть

Алгоритм волновой трассировки относится к числу классических методов решения задачи построения кратчайшего пути. Относится к классу алгоритмов поиска в ширину и используется при компьютерной разводке соединительных проводников на поверхности электронных микросхем [1].

Волновой алгоритм относится к классу алгоритмов «динамического программирования». Алгоритм состоит из двух этапов: «распространение волны» и «восстановление пути».

Для осуществления первого этапа вводится матрица, в которой элементы сопоставляются некоторым целым числам для каждой клетки

пространственной сетки, наложенной на рассматриваемую прямоугольную область. В клетку назначения записывается число «0». Остальные ячейки помечаются, как не пройденные волной. Ячейка «0» испускает волну первого поколения и записывает значение «1» в соседние доступные ячейки, которые ещё не были посещены. Каждое новое поколение волны: помечает ячейки как пройденные; порождает волну следующего поколения, добавляя в ячейки новой волны своё значение, увеличенное на «единицу».

Процесс останавливается при выполнении одного из следующих условий:

1. когда текущее положение (начало пути) было помечено как пройденное волной;
2. когда все доступные ячейки были посещены.

Если в процессе работы алгоритма начальная точка всё-таки была достигнута волной, то в массив сохраняется путь от текущего положения до клетки назначения.

Восстановление пути – это второй этап работы волнового алгоритма. Для его выполнения нужно поместить в путевой список начальную точку: текущее положение. Далее произвести несколько итераций: выбрать ячейку, соседнюю с последней в списке, т.е. такую, которая была бы помечена числом на единицу меньше, чем последняя в путевом списке. Продолжать процесс до тех пор, пока в путевом списке не окажется ячейка с пометкой «0».

Постановка задачи

Введём в рассмотрение двумерное клетчатое пространство, состоящей из «проходимых» и «непроходимых» клеток, обозначена клетка начала и клетка конца. «Непроходимые» клетки являются неподвижными препятствиями. Введём допущение – клетки исходной матрицы зададим квадратными. В дальнейшем рассматриваемое пространство можно называть картой.

Цель алгоритма – проложить кратчайший путь от клетки начала к клетке конца, если это возможно.

Путь между клеткой начала и клеткой конца – это упорядоченный набор клеток, в котором каждая последующая клетка является смежной с предыдущей. От старта во все направления распространяется волна, причем каждая пройденная волной клетка помечается как «пройденная».

Примем, что волна распространяется в четырёх направлениях. Путь не должен проходить сквозь неподвижные препятствия.

Волна движется, пока не достигнет клетки конца или пока не останется клеток, которых волна не прошла. Если волна прошла все доступные клетки, но так и не достигла клетки конца, значит путь от начала до конца проложить невозможно. После достижения волной конца прокладывается путь.

Игровые персонажи представляют собой изображения танков. Площадь игровых персонажей равна целой клетки пространства.

Будем считать, что между игровым персонажем и препятствием произошло столкновение, если существует хотя бы одна точка, которая принадлежит одновременно и персонажу, и препятствию.

Реализация алгоритма

Исходя из поставленной задачи был построен алгоритм, представленный на рисунках 1 и 2.

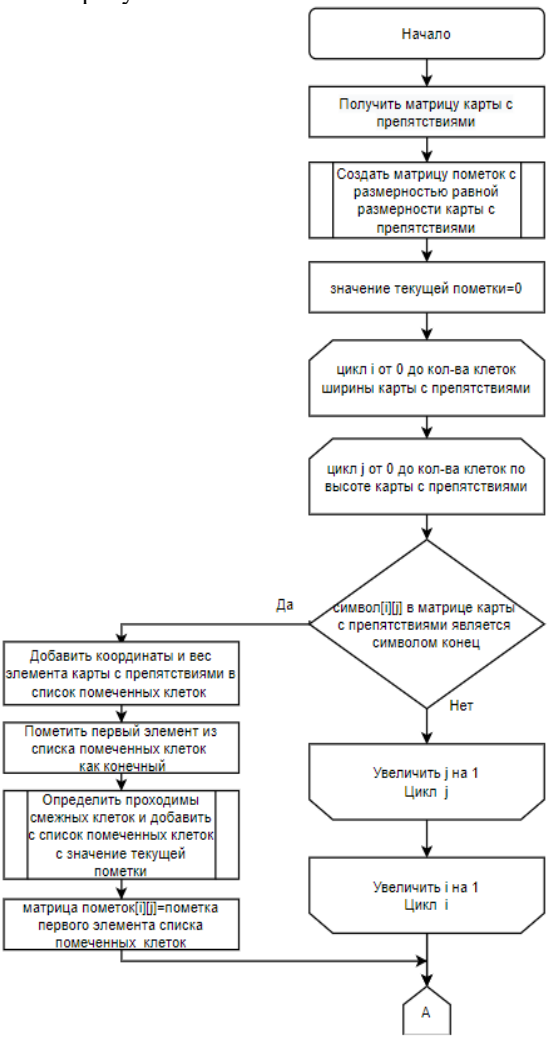


Рисунок 1 – Блок-схема алгоритма

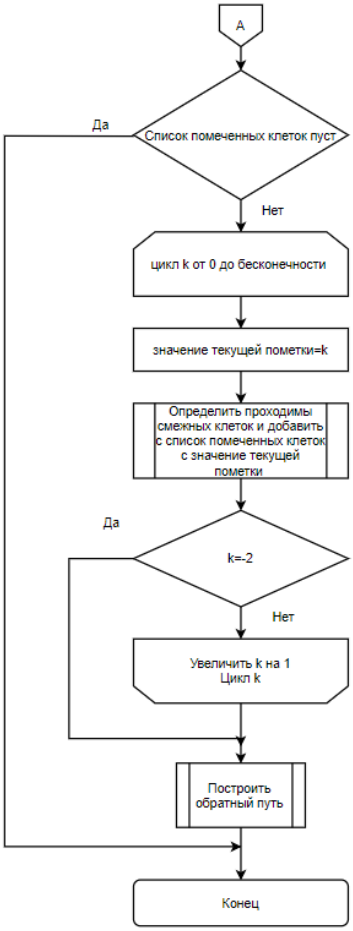


Рисунок 2 – Продолжение блок-схемы алгоритма

С помощью приведённого алгоритма была достигнута цель данной работы - реализовать алгоритм волновой трассировки для решения задачи нахождения кратчайшего пути между игровыми персонажами.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Тюкачев Н. А. C#. Алгоритмы и структуры данных: учебное пособие для СПО / Н. А. Тюкачев, В. Г. Хлебостроев. — Санкт-Петербург : Лань, 2021. — 232 с.

**УДК 519.812.3**

**Кузнецова К.И., Белов В.В.**

# АВТОМАТИЗАЦИЯ УПРАВЛЕНИЯ ПРОИЗВОДСТВОМ ИЗДЕЛИЙ ПРОМЫШЛЕННОГО ОБОГРЕВА И АНАЛИЗА ЕГО ЭФФЕКТИВНОСТИ В ООО «НПО РИЗУР»

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В работе рассматриваются вопросы разработки дополнительного программного обеспечения для автоматизированной системы ООО «НПО РИЗУР», позволяющего улучшить управление производством изделий промышленного обогрева.

В качестве объекта исследования рассматриваются автоматизированные системы общества с ограниченной ответственностью научно-производственного объединения РИЗУР (далее в тексте НПО РИЗУР), специализирующегося на производстве промышленных нагревательных систем. Предметом исследования являются вопросы проектирования, реализации и анализа эффективности дополнительных улучшающих решений для автоматизации управления производством изделий промышленного обогрева.

Сфера использования изделий промышленного обогрева достаточно широка: они используется во взрывоопасных и в общепромышленных зонах. Основными потребителями данных изделий являются предприятия нефтегазовой промышленности, составляющие важнейший сегмент промышленности России и оказывающие серьезное влияние на экономические, социально-политические и промышленные связи стран.

Оборудование промышленного обогрева позволяет решать на предприятиях следующие задачи:

- обеспечение необходимых климатических условий для надежной и безотказной работы оборудования;
- точное поддержание необходимой температуры в обогреваемом пространстве;
- защита от замерзания и образования конденсата;
- обогрев приборов и сохранение метрологических характеристик;
- подогрев нефти при низких температурах на выходе из устья скважин;
- местный обогрев участков трубопроводов.

**Платформа средств автоматизации НПО РИЗУР**

Основной системой для бизнеса являются ERP-инструменты. Система планирования ресурсов предприятия (ERP) — это набор модулей для бизнеса, предоставляющий инструменты для таких деятельностей, как бухгалтерский учёт, управление персоналом, цепочка поставок,

управление складом и многое другое. Данные всех применяемых модулей интегрируются в единую цифровую экосистему, что позволяет получить доступ ко всей этой информации из различных модулей, видеть разные отделы и выполнять более надёжный анализ данных. Такое решение даёт возможность проанализировать, как улучшения процессов и другие усилия по повышению эффективности, продаж или кадров влияют на доходы и рост. Многие из крупнейших и наиболее успешных компаний в мире используют инструменты ERP.

За период 2013–2019 годов разработчики ERP за довели автоматизацию производственных операций до такого уровня, что, во-первых, пользователь не может собрать технически некорректную номенклатурную позицию, а во-вторых, благодаря выходу на ограниченный номенклатурный ряд разработчикам удалось создать автоматизированные маршрутные карты на все номенклатурные позиции, то есть рабочий на производстве полностью понимает схему сборки изделия и этапы производственной цепочки. Процент брака вследствие такой автоматизации ничтожно мал, и основная причина его возникновения – банальная – плохое качество вспомогательных материалов или крепёжных элементов. И даже такую причину можно исключить, усилив входной контроль на предприятии.

На предприятии НПО РИЗУР используется система 1С:ERP, представляющая собой операционный блок, нацеленный на автоматизацию ключевых бизнес-процессов, оперативный контроль основных показателей деятельности предприятия, обеспечение взаимодействия между структурными подразделениями организации, координацию их работы и оценку эффективности.

Главным преимуществом 1С:ERP является то, что она представляет собой комплекс инструментов, способных связать воедино все процессы даже самого масштабного предприятия, при этом оптимизировав их схему, одновременно сделав их полностью прозрачными.

1С:ERP состоит из функциональных блоков, представляющих собой подсистемы, каждая из которых включает в себя совокупность инструментов и настроек, позволяющих осуществлять определённые функции. Для настройки 1С:ERP НСИ (Нормативно-Справочная Информация), настройки интеграции, а также для начального заполнения и администрирования в системе предусмотрен отдельный функциональный блок – «НСИ и администрирование».

Все настройки довольно гибкие, что позволяет адаптировать типовую конфигурацию 1С:ERP под предприятия с учётом всевозможных особенностей их деятельности.

В рамках НПО РИЗУР система 1С:ERP конфигурирована согласно стандартной логике ERP и не имеет глобальных отклонений в конфигурации. Это обуславливает то, что все задачи пользователей

можно штатно решить программными методами. Разумеется, всё же имеются некоторые отклонения от штатной конфигурации, а также содержатся отчеты и печатные формы, созданные под конкретные потребности конечных пользователей, но они незначительны.

**Сценарий управления производством НПО РИЗУР**

На предприятии НПО РИЗУР сценарий управления производством выглядит следующим образом:

1. Пользователь порождает потребность в производстве изделия.

2. Заказ на производство изделия попадает в очередь управления заказами.

3. Оператор или начальник производственно-диспетчерского отдела обрабатывает заказ: проверяет его на наличие ошибок, и затем, посредством создания ресурсной спецификации на изделие, где указаны все участвующие цеха производства, формирует производственные этапы.

4. Начальник производства и начальники цехов контролируют сроки изготовления изделия.

5. Отдел качества осуществляет проверку качества изготовленного изделия; в случае ненадлежащего качества или выявления несоответствий, изделие дорабатывается; в случае же надлежащего качества подготавливается паспорт и заверяется печатью ОТК.

6. По готовности изделие выпускается на склад – создаются документы передачи продукции и упаковочный лист, приходный ордер. Производственный цикл завершается.

**Необходимость разработки дополнительного ПО**

Необходимость разработки дополнительного программного обеспечения продиктована несколькими факторами:

1. «Гибкость» ценообразования в разрезе комплекса внешних экономических факторов.

2. Динамический расчет себестоимости продукции.

3. Сравнительный анализ по нормированию производственных операций, показывающий КПД сотрудников в заданном периоде.

4. Снижение производственных издержек.

Можно выделить следующие наиболее актуальные задачи по улучшению системы ERP на НПО РИЗУР.

**А. Повышение оперативности процесса формирования производственных этапов и выявление забытых заказов**

На исследуемом предприятии достаточно высок процент заказов на производство, где есть ошибки на входе. Заказы создаются в других отделах и затем передаются в производственно-диспетчерский отдел (ПДО) для формирования производственных этапов. Оператор ПДО проверяет полученные заказы и, при обнаружении ошибок, начинает принимать меры по их исправлению.

Виды ошибок разные – ненадлежащим образом оформлен заказ, технически некорректно собранная номенклатурная позиция, пропуск дополнительных реквизитов и прочее. В любом случае, оператору приходится тратить время на то, чтобы связаться с менеджером, который должен устранить ошибку.

Сократить время на формирование производственных этапов можно за счёт добавления в документ «Заказ на производство» специального признака ошибки с типом, выбираемым из списка ошибок. Этот признак позволит, во-первых, указать на сам факт запуска заказа с ошибкой, а во-вторых, отобразить суть допущенной ошибки. Информация, ассоциированная с типом ошибки, будет весьма полезна пользователям из ПДО, поскольку у почти всех этих пользователей нет доступа к версионированию документа, то есть, они не могут отследить, кто и когда вносил изменения в документ, и какими были эти изменения. Тип ошибки позволит специалистам ПДО быстро понять, какая конкретно ошибка была допущена при создании заказа на производство, и оперативно устранить её.

Установка признака ошибок – несложная, но очень действенная автоматизация, поскольку она помимо повышения оперативности исправления ошибок в запущенных заказах, решит ещё одну проблему – «забытые» заказы. Кроме того, она позволит формировать статистические данные по качеству формирования запускаемых заказов, в частности, определять количества некорректных запусков заказов на производство по отделам, типам продукции, временным интервалам. Даже распроведенный или закрытый заказ все равно сохраняет признак, так что пользователь, сделав отбор всего лишь по одному реквизиту, сможет быстро вывести список документов, в которых имеются проблемы. Алгоритм внедрения такой оптимизации имеет вид:

1. Редактируем список реквизитов.
2. Создаём дополнительный реквизитом с типом значения «Дополнительное значение».
3. Добавляем новые значения, по которым впоследствии можно будет делать отбор и сортировку.
4. Выводим значение данного реквизита в динамический список регистра.

**Б. Снижение брака и пересортицы**

Интеграция маршрутных листов для снижения брака и пересортицы. Написание по потребности пользователей специальных отчётов: «Прерванные и остановленные заказы», «Заводской номер», «Ведомость по товарам на складах», «Выполнение плана по количеству».

На предприятии не ведется учёт брака, кроме того, не заложено понятие пересортицы, которая выявляется при проведении инвентаризационных мероприятий и означает одновременный излишек и

недостачу товаров одного наименования, но разного сорта. Следовательно, возрастает объем номенклатуры по товарно-материальным ценностям (ТМЦ), которые числятся на остатках кладовых в производственных цехах. Из-за этого номенклатурный справочник разрастается и вести учёт становится сложнее, ведь линейка реализуемой продукции не расширяется, а количество материалов на складе увеличивается. Другими словами, одно и то же изделие возможно изготовить из разных комбинаций множества материалов. Пользователь, ведущий учёт (оператор), может начать путать материалы, что впоследствии выявит проведенная инвентаризация.

Для недопущения подобной ситуации предлагается автоматизировать систему закупок материалов, для этого необходимо разработать алгоритм по интеграции маршрутных карт на изделие в статистику по закупкам. Отдел снабжения, в случае невозможности закупки конкретного материала предлагает аналогичный. И именно этот аналог должен быть учтен в маршрутной карте на изделие во вкладке «Замена материалов». То есть база данных должна показать пользователю отдела снабжения варианты номенклатур, которые уже фигурируют в смежных регистрах.

Отчет «Ведомость по товарам на складах» отражает действительный остаток ТМЦ на складе в указанном периоде, указывает пользователю на корректность или некорректность отражения приобретений ТМЦ в разрезе кладовой цеха или склада через начальный и конечный остаток.

**В. Анализ причин нарушений сроков выпуска продукции**

Информацию для выяснения причин невыполнения производственного плана поможет сформировать «Отчёт по прерванным и остановленным заказам». Сведения этого отчёта дадут пользователям возможность проанализировать причины невыполнения производственного плана, поскольку количество прерванных и остановленных заказов напрямую сказывается на сроках исполнения производственного задания.

Отклонение по выпуску продукции от намеченного плана будет отражать отчёт «Выполнение плана по количеству» – один из основных отчётов производства.

Поиск всей информации по заказу, в котором фигурировало рекламационное изделие, предоставит отчёт «Заводской номер», обеспечивающий быстрый поиск изделия по заводскому номеру.

**Г. Формирование маршрутных карт на изделие с минимальными временными затратами**

Производство берет в работу только те заказы клиента, где номенклатурная карточка принадлежит одной из групп видов

номенклатуры, которые выпускаются на рассматриваемом нами предприятии.

На каждое изделие производственно-диспетчерским отделом создаётся своя ресурсная спецификация, где описан путь изделия по этапам производства с учетом используемых материалов и трудозатрат (видов работ). В качестве средства автоматизации в свое время была разработана группа ресурсных спецификаций под названием «Шаблонирование». На основании таких ресурсных спецификаций пользователь может создать маршрутную карту на изделие с минимальными временными затратами.

По факту проведённого исследования, исходя из того, что реализация поставленных задач снижает человеческий фактор и дает возможность анализировать состояние производства, можно сделать вывод, что внедрение предложенных решений по оптимизации автоматизации управления производства изделий промышленного обогрева на НПО РИЗУР позволит повысить производительность и конкурентоспособность компании, а также проводить анализ эффективности для дальнейшего построения стратегии производства.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Учебное электронное текстовое издание «Производственный менеджмент: от теории к практике», проф., д-р экон. наук Л.Д. Гительман, Екатеринбург, 2017.

2. Статья «Формирование и развитие процесса управления производством на предприятии», проф., д-р экон. наук Б.Н. Герасимов, Самара, 2020.

3. Статья «60 важнейших статистических моментов в автоматизации процессов», Oracle, 2022.

Интернет-ресурсы:

1. Компания ООО «НПО РИЗУР»: https://rizur.ru/.

2. Система программ 1С:Предприятие: https://v8.1c.ru/

3. 1С:ERP: https://v8.1c.ru/erp/poleznye-materialy/books/.

**УДК 004.658**

**Мосякин Д.Д.**

# ИССЛЕДОВАНИЕ СПОСОБОВ И АЛГОРИТМОВ АВТОМАТИЗАЦИИ ПРОЦЕССА УПРАВЛЕНИЯ АВТОПАРКОМ ОРГАНИЗАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

Статья посвящена обзору способов автоматизации процесса управления автопарком предприятия. Флит-менеджмент был рассмотрен как один из способов автоматизации управления. Также был разработан алгоритм для управления автопарком организации.

Когда компания отказывается от аренды автомобилей и формирует собственный автопарк, уходят одни проблемы и появляются другие. С одной стороны, она перестаёт зависеть от сторонних организаций, к ней переходит полный контроль над техникой и людьми. Но с другой, компании приходится самостоятельно решать вопросы на расстоянии, когда автомобиль находится за сотни километров от офиса. По мере своего развития каждая компания имеющая личный автопарк задумывается о способах повышения эффективности его работы. Одним из таких способов является fleet-management.

Fleet management - область управления, которая занимается организацией работы автопарка.

Цель работы - исследование и разработка алгоритма, позволяющего автоматизировать процесс управления автопарком организации.

Для достижения поставленной цели, были выделены следующие задачи:

1. Исследовать задачи флит-менеджмента.

2. Разработать алгоритм автоматизации управления автопарком.

Задачами флит-менеджмента являются:

1. Отвечать за бесперебойную работу техники. Менеджер должен налаживать быстрый ремонт автомобилей, обеспечивать своевременное ТО и страховку, заменять машины и пополнять автопарк по запросу руководства.

2. Оптимизировать работу сотрудников: водителей, техников и других специалистов, связанных с эксплуатацией автомобилей. Флит-менеджер налаживает работу так, чтобы не было простоя ни техники, ни персонала.

3. Оптимизировать затраты на обслуживание авто.

4. Работать с подрядчиками. Вести документацию: подписывать договоры, оформлять оплату заказов, готовить закрывающие документы, отправлять документацию по почте.

5. Предоставлять отчёты о результатах грузоперевозок: о расходах на топливо, работе водителей, сроках доставки грузов, решении технических проблем.

В целом, флит-менеджер должен обеспечить бесперебойную работу автопарка, совершенствовать его и оптимизировать расходы на эксплуатацию техники.

Часто управление автопарком предприятия оказывается сосредоточено на одной задаче, а на другие вопросы внимания уделяется слишком мало. Это влечёт лишние траты и падение доходов. Такая ситуация недопустима, так как перекос в управлении постепенно снизит эффективность всего подразделения.

В общем случает взаимодействие сотрудников с автопарком организации можно представить на схеме информационных потоков ( рисунок 1).

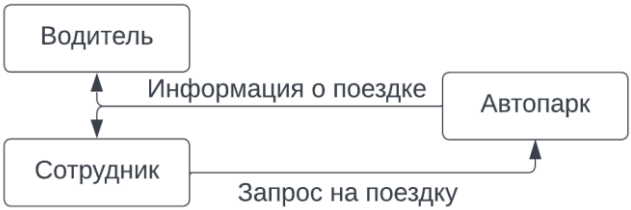


Рисунок 1 - Схема информационных потоков

Основной алгоритм управления автопарком можно реализовать с помощью системы 1С: Предприятие.

Составляем документ на основе справочников с информацией о водителях и сотрудниках. Если сотруднику потребуется использовать автомобиль, то необходимо создать новую запись, которая будет содержать такую информацию как дата запланированной поездки, её время, причина и степень важности. Степень важности отвечает за то, какую поездку отменить, а какую оставить в чрезвычайной ситуации, когда срочно требуется автомобиль.

В итоге благодаря подобной системе предприятие избежит ряда проблем, связанных с использованием автопарка. Персонал, которому необходим транспорт по каким-либо причинам будет иметь возможность забронировать его заранее, благодаря чему избежит ряда непредвиденных ситуаций. Также был предусмотрен такой вариант событий в котором двум сотрудникам придётся решать кто поедет на машине предприятия, а кто отменит встречу либо поедет сам.

В данной работе были рассмотрены способы и алгоритмы управления автопарком и как они влияют на эффективность организации. Был спроектирован простейший алгоритм управления автопарком организации.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Хвостикова В.А., Китаева И.С. Организация управления ресурсами и затратами 2019. 44 с

2. H. Chisom Resource Management: Definition, Importance, and Planning

3. Терехова А.Е., Козорез М.С. Показатели эффективности управления автопарком 2013. 51с

**УДК 004.056.5**

**Пасынков М. И., Швечкова О.Г.**

# СРАВНИТЕЛЬНЫЙ АНАЛИЗ ПРОГРАММНОЙ РЕАЛИЗАЦИИ СХЕМЫ ЭЛЕКТРОННОЙ ЦИФРОВОЙ ПОДПИСИ НА БАЗЕ ОТЕЧЕСТВЕННОГО АЛГОРИТМА ГОСТ В РАМКАХ ДИСЦИПЛИНЫ «ЗАЩИТА ИНФОРМАЦИИ»

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

Целью данной работы является сравнение программной реализации на языках Python и C# схемы электронной цифровой подписи отечественного стандарта ГОСТ Р 34.10-2012 как одного из видов криптографических методов защиты информации в рамках учебной дисциплины «Защита информации».

Как известно, в настоящее время механизм использования электронной цифровой подписи(ЭЦП) является одним из надёжных способов обеспечения таких важных элементов реализации всех видов электронных транзакций и системы электронного документооборота как целостность и секретность передаваемой информации. Именно поэтому изучение особенностей отдельных видов алгоритмов ЭЦП и сравнение их программной реализации определяет актуальность темы исследования.

Электронная цифровая подпись представляет собой относительно небольшое количество дополнительной цифровой информации, передаваемой вместе с подписываемым текстом.

Согласно стандартной процедуре механизм электронной цифровой подписи включает два этапа: этап постановки или формирования ЭЦП и этап проверки подлинности зашифрованного сообщения, подтверждённого ЭЦП.

В процедуре постановки подписи используется секретный ключ отправителя сообщения, в процедуре проверки подписи – открытый ключ отправителя.

При формировании ЭЦП отправитель прежде всего вычисляет хеш-функцию h(M) подписываемого текста M. Вычисленное значение хеш-функции h(M) представляет собой один короткий блок информации m, характеризующий весь текст M в целом. Затем число m шифруется секретным ключом отправителя. Получаемая при этом пара чисел представляет собой ЭЦП для данного текста M.

При проверке ЭЦП получатель сообщения вновь вычисляет хеш-функцию m=h(M) принятого по каналу текста M, после чего при помощи открытого ключа отправителя проверяет, соответствует ли полученная подпись вычисленному значению m хеш-функции.

Принципиальным моментом в системе ЭЦП является невозможность подделки ЭЦП пользователя без знания его секретного ключа подписания.

В работе рассмотрены реализации алгоритма электронной цифровой подписи, соответствующего ГОСТ Р 34.10-2012 [1] на языках Python и C#.

Основная сложность реализации данного алгоритма состоит в использовании функции «Стрибог» для вычисления хэш-значений [2]. Эта функция разбивает исходное сообщение на последовательности длиной 512 бит, применяет к каждой части раундовую функцию сжатия, которая, в свою очередь, содержит применение следующих операций:

- нелинейное преобразование, при котором к каждому байту состояния (исходные 512 бит рассматриваются как массив 8x8 байт) независимо применяется табличная замена;
- переупорядочивание байт состояния;
- линейное преобразование, при котором исходные 512 бит рассматриваются как 8 64-битных вектором; каждый из этих векторов заменяется результатом умножения на определенную стандартом матрицу 64x64.

При этом функции сжатия на каждом шаге передается не только часть сообщения длиной 512 бит, но и результат сжатия предыдущей части сообщения. Поэтому обработка разных частей сообщения не может осуществляться параллельно.

Далее происходит сложение значений функции сжатия для каждой части сообщения по модулю $2^{512}$. Результат сложения передается в функцию, вычисляющую хэш-значение.

Кроме вычисления хэш-значения, сложность представляет выбор коэффициентов, задающих эллиптическую кривую, на которой происходит шифрование. Эти коэффициенты включают:

p – простое число, размер конечного поля;

a, b – коэффициенты уравнения эллиптической кривой;

G – базовая точка, генерирующая подгруппу;

n – порядок подгруппы;

h – кофактор подгруппы.

Выбор коэффициентов эллиптической кривой непосредственно влияет на надежность шифрования, при этом генерация случайных коэффициентов не может считаться эффективной. Так, кривые, для которых порядок конечного поля равен порядку самой кривой (1).

$$p = hn, \quad (1)$$

Сингулярные эллиптические кривые, для которых выполняется равенство (2), также не могут использоваться для шифрования, так как в этом случае для задачи дискретного логарифмирования существуют субэкспоненциальные методы решения, то есть задача нахождения дискретного логарифма на эллиптической кривой может быть сведена к задаче нахождения дискретного логарифма в конечном поле.

$$\left(\frac{a}{3}\right)^3 + \left(\frac{b}{2}\right)^2 = 0 \quad (2)$$

Для сравнения полученных алгоритмов предлагается сформировать цифровую подпись для файлов разных размеров, оценить время формирования подписи и объем памяти, используемой программой. Результаты эксперимента представлены в таблице 1.

Таблица 2 – Результаты замеров времени и объёма памяти

| Размер файла | Python | | C# | |
|---|---|---|---|---|
| | Время выполнения, сек | Объем памяти Мб | Время выполнения, сек | Объем памяти Мб |
| 1 Кб | 0.2 | 14 | 0.23 | 9 |
| 100 Кб | 1.32 | 14 | 0.58 | 10 |
| 1 Мб | 12.19 | 16 | 3.65 | 13 |
| 10 Мб | 326.05 | 34 | 33.27 | 45 |

Исходя из данных таблицы, время формирования подписи на языке Python оказывается значительно больше аналогичного параметра для программы на C#. При этом большая часть времени затачивается на разбиение сообщение на фрагменты и вычисление хэш-значений для этих фрагментов.

По объёму памяти программа на Python оказывается эффективнее программы на C# при формировании подписи для файлов, больших 1 Мб.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. ГОСТ Р 34.10-2012 Информационная технология. Криптографическая защита информации. Процессы формирования и проверки электронной цифровой подписи: национальный стандарт Российской Федерации: дата введения 2012-08-07 / Центр защиты информации и специальной связи ФСБ России. – Изд. официальное. – Москва: Стандартинформ, 2012. – 29 с.

2. ГОСТ Р 34.11-2012 Информационная технология. Криптографическая защита информации. Функция хэширования: национальный стандарт Российской Федерации: введен 2012-08-07 / Центр защиты информации и специальной связи ФСБ России. – Изд. официальное. – Москва: Стандартинформ, 2012. – 34 с.

3. Р 50.1.114-2016 Информационная технология. Криптографическая защита информации. Параметры эллиптических кривых для криптографических алгоритмов и протоколов: рекомендации по стандартизации: дата введения 2016-11-28 / Технический комитет по стандартизации ТК 26 «Криптографическая защита информации». – Изд. официальное. – Москва: Стандартинформ, 2016. – 11 с.

4. FIPS PUB 186-4 Computer Security. Criptography. Digital Signature Standard (DSS) : Federal Information Processing Standards Publication : July 2013 / Information Technology Laboratory, National Institute of Standards and Technology. – 121 p.

5. Tan, Ch. K. (28.09.2002). C# BigInteger Class [Исходный код]. https://www.codeproject.com/Articles/2728/C-BigInteger-Class

**УДК 004.932.72'1**

**Пасынков М.И., Цуканова Н.И.**

# АЛГОРИТМЫ SORT И DEEPSORT ОТСЛЕЖИВАНИЯ ОБЪЕКТОВ НА ВИДЕОЗАПИСИ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

Рассматриваются алгоритмы SORT и DeepSORT, использующиеся для отслеживания объектов на видеозаписи в режиме реального времени. Проводится сравнительный анализ данных алгоритмов.

Отслеживанием объектов называется процесс определения количества и состояния множества объектов на основе данных, полученных с некоторых датчиков, например, видеокамеры [2].

В данной статье предлагается сравнить алгоритмы SORT и DeepSORT для отслеживания объектов на видеозаписи. Использование данных алгоритмов предполагает, что видеозапись, подаваемая на вход алгоритма, уже прошла процесс распознавания объектов. То есть для каждого кадра видеозаписи уже определены прямоугольные области, в которых находятся искомые объекты. Задача отслеживания состоит в том, чтобы связать одни и те же объекты на разных кадрах видеозаписи.

При использовании алгоритма SORT [1] распознаваемые объекты характеризуются динамикой движения. Эта характеристика складывается из следующих показателей:

- координаты центра прямоугольника, которым выделяется объект;
- высота этого прямоугольника;
- соотношение сторон этого прямоугольника;
- производные от каждой из предыдущих величин.

По этим показателям для каждого объекта, который выделен прямоугольной областью на текущем кадре, предсказывается положение прямоугольной области на следующем кадре. Для предсказания используется фильтр Калмана [3, 4].

Далее спрогнозированные для следующего кадра положения прямоугольных областей сопоставляются с действительно найденными на этом кадре прямоугольными областями. Оптимальный вариант сопоставления определяется с помощью Венгерского метода [6].

Таким образом, происходит связывание объектов на одном кадре с предположительно теми же объектами на следующем кадре. При этом не используются данные о положении объектов на всех предыдущих кадрах, что позволяет выполнять отслеживание в режиме реального времени.

Основной недостаток алгоритма SORT заключается в том, что при исчезновении объекта из поля зрения и повторном его появлении через некоторое время ему приписывается новый идентификатор. То есть алгоритм SORT не позволяет длительно хранить в памяти объекты.

При использовании алгоритма DeepSORT [5], помимо динамики движения, объект характеризуется внешним видом, который составляется из 128 показателей. Для получения этих показателей изображение объекта передаётся на вход нейронной сети, архитектура которой представлена на рисунке 1.

| Name | Patch Size/Stride | Output Size |
|---|---|---|
| Conv 1 | $3 \times 3/1$ | $32 \times 128 \times 64$ |
| Conv 2 | $3 \times 3/1$ | $32 \times 128 \times 64$ |
| Max Pool 3 | $3 \times 3/2$ | $32 \times 64 \times 32$ |
| Residual 4 | $3 \times 3/1$ | $32 \times 64 \times 32$ |
| Residual 5 | $3 \times 3/1$ | $32 \times 64 \times 32$ |
| Residual 6 | $3 \times 3/2$ | $64 \times 32 \times 16$ |
| Residual 7 | $3 \times 3/1$ | $64 \times 32 \times 16$ |
| Residual 8 | $3 \times 3/2$ | $128 \times 16 \times 8$ |
| Residual 9 | $3 \times 3/1$ | $128 \times 16 \times 8$ |
| Dense 10 | | 128 |
| Batch and $\ell_2$ normalization | | 128 |

Рисунок 1 — Архитектура нейронной сети

После этого определяется расстояние между спрогнозированными положениями объектов, объектами на предыдущем кадре и действительно найденными на следующем кадре объектами. Это расстояние определяется по формуле (1).

$$D = \lambda * D_k + (1-\lambda) * D_a, \quad (1)$$

Здесь $\lambda$ – некоторая константа, которая показывает, какая из двух характеристик объекта (динамика движения или внешний вид) обладает большей значимостью. Чем больше $\lambda$, тем большее значение будет иметь динамика движения при определении схожести объектов. И наоборот, чем меньше $\lambda$, тем большее значение будет иметь внешний вид, как характеристика искомых объектов.

Переменная $D_k$ обозначает схожесть предсказанного фильтром Калмана объекта с реально найденным на кадре объектом. Эта схожесть определяется как коэффициент Жаккара (2), то есть отношение пересечения прямоугольных областей этих объектов к объединению этих прямоугольных областей.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \quad (2)$$

Переменная $D_a$ обозначает схожесть внешнего вида объекта на предыдущем кадре с внешним видом объекта на следующем кадре. Метрикой схожести в этом случае выступает косинусное расстояние (3) между векторами, состоящими из 128 показателей внешнего вида объекта.

$$similarity = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}} \quad (3)$$

После определения расстояния объекты на следующем кадре сопоставляются с объектами на предыдущем кадре с использованием Венгерского метода аналогично алгоритму SORT.

Таким образом, алгоритм DeepSORT также позволяет отслеживать объекты на видеозаписи, но, благодаря введению показателей внешнего вида, идентификаторы объектов «перезаписываются» в 2 раза реже [5]. То есть объекту реже приписывается новый идентификатор в случаях его исчезновения с кадра или резком изменении положения объекта на кадре.

Однако введение дополнительной характеристики объекта, которая рассчитывается с помощью нейронной сети, увеличивает время обработки видеозаписи в 1.5 раза. Иными словами, повышение точности работы алгоритма DeepSORT в сравнении с алгоритмом SORT приводит к увеличению его трудоемкости. Поэтому выбор алгоритма зависит от требований, предъявляемых к конкретной задаче.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Bewley A., Ge Z., Ott L., Ramos F., Upcroft B. Simple online and realtime tracking. 2016 IEEE International Conference on Image Processing (ICIP). – 2016. – С. 3464-3468.

2. Granstrom K., Baum M., Reuter St. Extended Object Tracking: Introduction, Overview and Applications. Journal of Advances in Information Fusion. – 2017.

3. Kalman R. E. F new approach to linear filtering and prediction problems // Journal of basic Engineering. – 1960. – Т. 82. №. 1. – С. 35-45.

4. Shrinivasan Sh. The Kalman Filter: An algorithm for making sense of fused sensor insight // Towards Data Science [сайт]. – 2018. – URL: https://towardsdatascience.com/kalman-filter-an-algorithm-for-making-sense-from-the-insights-of-various-sensors-fused-together-ddf67597f35e. – Дата публикации: 18 апреля 2018.

5. Wojke N., Bewley A., Paulus D. Simple online and realtime tracking with a deep association metric. 2017 IEEE International Conference on Image Processing (ICIP). – 2017. – С. 3645-3649.

6. Асанов, М.О. – Дискретная математика: графы, матроиды, алгоритмы / М. О. Асанов, В. А. Баранский, В. В. Расин. – Ижевск: НИЦ «РХД», 2001. – 288 стр.

**УДК 004.421**

**Проказникова Е.Н., Анастасьев А.А.**

# ИСПОЛЬЗОВАНИЕ АЛГОРИТМА МАЙЕРСА ДЛЯ ОПТИМАЛЬНОГО ПРЕОБРАЗОВАНИЯ СПИСКОВ В ВЕБ-ПРИЛОЖЕНИЯХ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В статье рассматривается алгоритм Юджина Майерса для эффективного преобразования цепочек символов, а также обсуждается применимость данного алгоритма к разработке веб-приложений.

На данный момент одной из наиболее популярной разновидностью программного обеспечения являются веб-приложения.

Как правило веб-приложения нужны для предоставления пользователю какой-либо информации или услуг. Так или иначе, практически в каждом веб-приложении происходит отображение наборов данных, элементы которых идентичны по своей структуре, или, иначе говоря, отображение списков данных. Примерами могут быть доступные для покупки товары из каталога интернет-магазина, детальный почасовой прогноз погоды, список сайтов в результате поискового запроса в браузере, отзывы на блюдо на сайте ресторана, список диалогов пользователя в социальной сети и так далее.

В большинстве случаев списки данных необходимо периодически обновлять и отображать новую актуальную информацию. Например, если при просмотре писем в почтовом клиенте пришло новое сообщение.

В таких случаях возникает ситуация, когда необходимо заменить один список идентичных по своей структуре данных другим списком новых данных с такой же структурой. Самым простым подходом в таком случае является удаление старого списка и отображение нового, более актуального.

При таком подходе возникает проблема. Если количество изменений в списке мало, например, произошло добавление или удаление всего одного элемента, то остальные данные, состояние которых не изменилось, все равно будут удалены и отображены заново. Это влечет за собой неэффективный расход мощностей пользовательского устройства, и, следовательно, увеличивает время отклика приложения, а также пользовательский опыт взаимодействия с ним.

В том случае, когда отображение элементов данных имеет простой дизайн, например, состоит только из текстовых полей, полное обновление списка будет происходить быстро. Тогда нерациональным расходом мощностей пользовательских устройств можно пренебречь. Однако бывают случаи, когда элементы списка имеют сложный дизайн и время на их отображение является существенным. В такой ситуации повторное

отображение большого количества неизменившихся элементов является не лучшим решением.

Рассмотрим, какие подходы к этой проблеме существуют в веб-разработке на данный момент.

1. Пагинация - это порядковая нумерация страниц, которая применяется для разделения ссылочных блоков на веб-сайте. Это такой способ отображения большого массива данных, когда единовременно доступно лишь ограниченное количество элементов. Для получения следующей или предыдущей порции данных необходимо выполнить повторный запрос. В таком случае при обновлении списка повторно отобразятся только доступные на данный момент элементы, то есть их количество заранее ограничено. Преимуществом данного метода является простота реализации. Кроме того, пагинация может использоваться для упрощения поиска нужного элемента (товара в каталоге, статьи на информационном ресурсе и пр.). Недостатком является то, что список отображается не полностью, а также неудобство использования (пользователю нужно переключаться на следующую страницу вручную).

2. Ограничение времени обновления данных. При таком подходе список данных обновляется не при всяком изменении, а один раз в определенный промежуток времени. Это позволяет контролировать общее количество отображений элементов данных. Недостатком является то, что приложению по-прежнему приходится обновлять список полностью, хоть и с меньшей частотой. Кроме того, этот способ не подходит в тех случаях, когда важна актуальность и скорость обновления информации, например, при отображении цен на валютной бирже.

Описанные подходы не решают проблему обновления списков с малым количеством изменений. Они лишь снижают негативные эффекты от её воздействия.

В связи с этим встает задача создания инструмента для оптимального преобразования идентичных по структуре списков данных в веб-приложениях.

В 1986 году Юджин Майерс, американский учёный в области информатики и биоинформатики, опубликовал статью под названием «Разностный алгоритм O(ND) и его вариации». Этот алгоритм позволяет найти оптимальную последовательность преобразований над одной цепочкой символов, чтобы получить вторую. Сложность данного алгоритма составляет O(N+D^2), где N – это сумма длин двух цепочек символов, а D – размер минимальной последовательности преобразований [1].

Данный алгоритм уже используется для решения аналогичной задачи. В 2017 году компания Google разработала инструмент для Android разработки под названием DiffUtil.

DiffUtil - это служебный класс, который вычисляет разницу между двумя списками и выводит последовательность операций обновления, преобразующих первый список во второй. Он использует разностный алгоритм Майерса для вычисления минимального количества обновлений для преобразования списков [2].

В официальном описании DiffUtil сказано, что его внутренняя реализация алгоритма Майерса оптимизирована по памяти и использует пространство O(N), чтобы найти минимальное количество операций добавления и удаления между двумя списками. При этом ожидаемая производительность по времени также составляет O(N+D^2).

В DiffUtil есть возможность особым образом обрабатывать перемещения элементов списка. Если включено обнаружение перемещения, это занимает дополнительно O(MN) времени, где M - общее количество добавленных элементов, а N - общее количество удаленных элементов. Если порядок элементов не менялся, есть возможность отключить обнаружение перемещений для повышения производительности.

Как следует из представленного описания, инструмент DiffUtil, разработанный компанией Google, использует алгоритм Майерса для оптимального преобразования списков в мобильных приложениях на платформе Android. Данная задача полностью соответствует нашей поставленной задаче, но реализована для мобильной платформы, что не удивительно, поскольку вопрос об оптимизации ресурсов пользовательского устройства в мобильной разработке стоит более остро, чем в разработке веб-приложений.

Таким образом, алгоритм Майерса находит кратчайший сценарий преобразований одной цепочки символов в другую. При этом он оптимизирован по памяти и занимает линейное пространство. Временная сложность алгоритма составляет O(N+D^2), которая при малом количестве преобразований D сводится к линейной O(N).

Более того, алгоритм Майерса используется компанией Google для решения аналогичной проблемы в области мобильной разработки для платформы Android.

Исходя из этого, алгоритм Майерса подходит для поставленной задачи эффективного преобразования списков в веб-приложениях.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Eugene, W., Myers. (1986). An O ( ND ) difference algorithm and its variations. Algorithmica, 1(1), 251-266. doi: 10.1007/BF01840446

2. DiffUtil [Электронный ресурс] / Режим доступа: https://developer.android.com/reference/androidx/recyclerview/widget/DiffUtil

**УДК 005.92**

**Пукин А.А.**

# ИССЛЕДОВАНИЕ И РАЗРАБОТКА АЛГОРИТМОВ ПРОЦЕССА СОГЛАСОВАНИЯ ЭЛЕКТРОННОГО ДОКУМЕНТООБОРОТА ДЛЯ ФИНАНСОВО-ТЕХНОЛОГИЧЕСКОЙ ОРГАНИЗАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

Важной частью электронного документооборота является согласование документов. В наше время компании все чаще и чаще переходят с ручного документооборота на электронный. Цель настоящей работы - автоматизировать существующий процесс согласования под современные реалии, учитывая условия, поставленные компанией.

Внедрение системы электронного документооборота актуально для любого учреждения. Внедрение системы повысит уровень контроля выполнения сотрудниками своих обязанностей, повысится исполнительская дисциплина, а вместе с этим организация приобретает:

1. Прозрачность движения документов.

2.Единое информационное пространство, информационно связывающее подразделения Департамента и контрагентов.

3. Гарантированное наличие информации и моментальный доступ к ней.

4. Ускоренный обмен документами и сроки исполнения в виду отсутствия необходимости в физическом размножении и перемещении документов.

5. Управляемые и контролируемые информационные потоки и как следствие повышение управляемости организацией в целом.

6. Электронного согласование документов будет в ускоренном режиме с сокращением сроков подписания.

Компания “Тинькофф” – одна из тех компаний, кто внедряет у себя систему электронного документооборота. В данный момент им требуется разработать систему согласования ЭДО.

Цель работы – автоматизировать процесс согласования в электронном документообороте с поставленными условиями от компании.

Поставленные задачи:

1. Спроектировать базу данных для хранения и обработки информации процесса согласования.

2. Изучить информацию по программе “Camunda” [1] для реализации BPMN схем процессов, показывающих наглядно работу процесса согласования.

3. Построить алгоритм процесса согласования.

Рассмотрим пример организационной структуры в компании на рисунок 1. Согласование происходит по подстатьям, указанным в счёте.

То есть чтобы согласовать весь счёт, нужно согласовать все подстатьи в нём. Каждая подстатья привязана к одному из главных отделов компании. Следовательно, при согласовании счёта берётся отдел каждой подстатьи и руководителю этого отдела отсылается счёт на согласование.

Подотделы нужны для того, чтобы освобождать главные отделы от согласований мелких или не самых важных счетов. Например, руководитель главного отдела может сделать ответственным своей подчинённый подотдел за счета до 1000000 рублей, а подчинённого своего подчинённого сделать ответственным за счета до 500000 рублей. И тогда при согласовании подстатей до этих сумм согласование до главного отдела не дойдёт, т.к. его согласуют его подчинённые, которых он назначил.

Также можно назначить сразу нескольких подчинённых одного уровня за одну и ту же подстатью. Это называется параллельное согласование. Здесь неважно, кто именно согласует. Главное, чтобы согласовал хотя бы один и согласование пойдёт дальше к следующим согласующим.

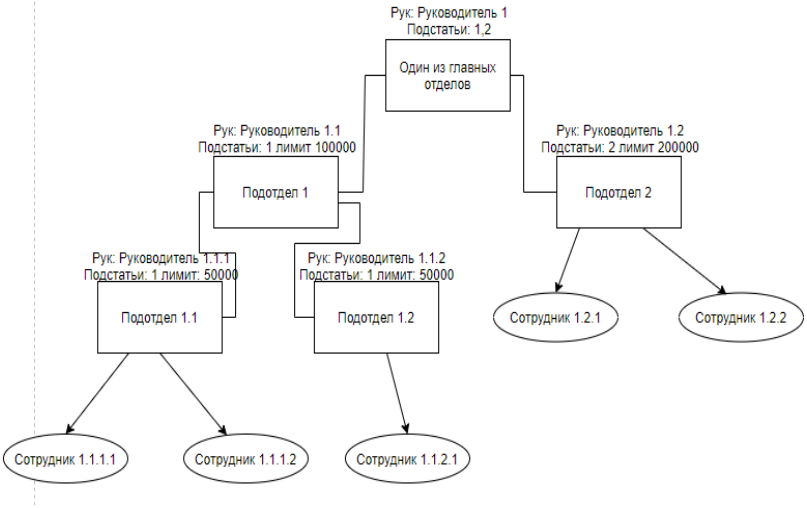


Рисунок 1 – Пример организационной структуры компании

Фрагмент алгоритма процесса согласования представлен на рисунок 2.

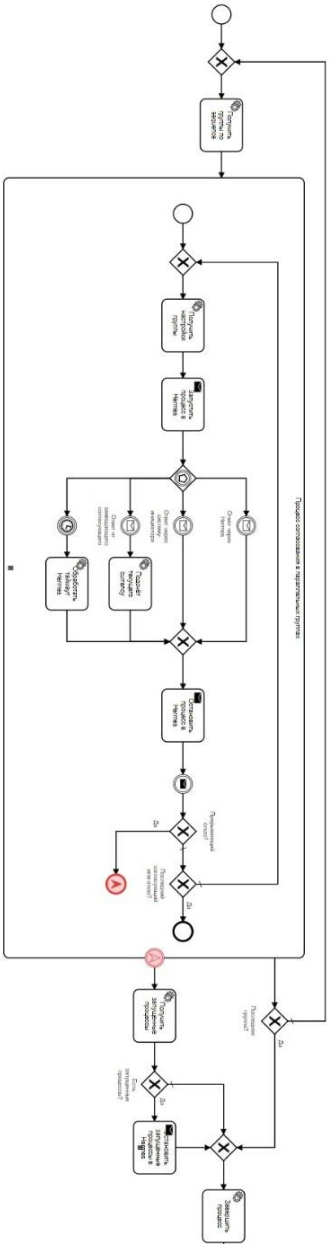

Рисунок 2 – Фрагмент алгоритма процесса согласования

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Camunda. Сайт приложения Camunda. – URL: https://camunda.com (дата обращения: 01.12.2022 г.).

**УДК 004.021**

**Ратников Д.В., Крошилина С.В.**

# ИСПОЛЬЗОВАНИЕ СПУТНИКОВОЙ СИСТЕМЫ ДИФФЕРЕНЦИАЛЬНОЙ КОРРЕКЦИИ ДЛЯ ПОВЫШЕНИЯ ТОЧНОСТИ ОПРЕДЕЛЕНИЯ МЕСТОПОЛОЖЕНИЯ ОБЪЕКТОВ НА МЕСТНОСТИ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В статье рассматривается работа алгоритма дифференциальной корректировки местоположения объекта на местности.

Одним из способов повышения точности определения местоположения объектов на местности является использование алгоритмов дифференциальной корректировки. Кроме повышения точности, улучшаются такие характеристики работы навигационной системы, как надежность, доступность и достоверность.

Спутниковая система дифференциальной коррекции [1] (SBAS — Satellite Based Augmentation System) — это геосинхронные спутниковые системы, предоставляющие услуги по повышению точности, целостности и доступности основных сигналов GNSS. Спутниковые вспомогательные системы поддерживают увеличение точности сигнала за счет использования спутниковой трансляции сообщений. Такие системы обычно состоят из нескольких наземных станций, координаты расположения которых известны с высокой степенью точности.

Суть большинства методов дифференциальной коррекции заключается в учете навигационной аппаратурой различного рода поправок, получаемых из альтернативных источников. Для различного рода применений источниками корректирующей информации являются УССИ (унифицированные станции сбора измерений), опорные координаты которых известны с высокой точностью. Как правило методы дифференциальной коррекции обеспечивают поправками ограниченную территорию Земли.

Рассмотрим принцип работы [3] дифференциальной коррекции:

1. Базовые станции мониторинга системы с заранее заданными координатами ведут непрерывное слежение за космической группировкой спутников.

2. Станции мониторинга системы передают накопленную информацию на контрольно-вычислительные станции (мастер-станции) системы.

3. На мастер-станциях определяются погрешности и формируются дифференциальные поправки на некую ограниченную территорию.

4. Вычисленные поправки отправляются на станции передачи данных, которые равномерно расположены на обслуживаемой территории.

5. Оттуда поправки передаются на геостационарные спутники.

6. Со спутников поправки отправляются конечному пользователю.

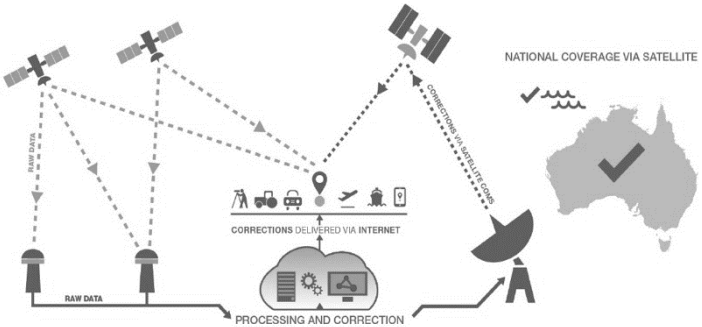


Рисунок 1 – Иллюстрация работы системы дифференциальной коррекции

SBAS уже реализована различными организациями [2] по всему миру и вот некоторые из них:

1. Wide Area Augmentation System (WAAS). Федеральное авиационное управление США разработало систему для предоставления поправок GPS и сертифицированного уровня целостности для авиационной отрасли, чтобы позволить самолетам выполнять точные заходы на посадку в аэропортах. Исправления также доступны бесплатно гражданским пользователям в Северной Америке.

2. European Geostationary Navigation Overlay Service (EGNOS).Европейское космическое агентство в сотрудничестве с Европейской комиссией разработало Европейскую геостационарную службу навигационного наложения, которая повышает точность определения местоположения, полученного с помощью GPS. сигналов и предупреждает пользователей о надежности сигналов GPS.

3. Система дифференциальной корректировки и мониторинга (СДКМ). В России также разрабатывают систему для повышения точности и контроля целостности навигационных систем ГЛОНАСС и GPS.

Без поправки гражданский сигнал GPS имеет точность около 10 метров. Приемники с поддержкой WAAS дифференциально корректируют большую часть этой ошибки в режиме реального времени, сравнивая скорректированный сигнал WAAS с входящим сигналом GPS и устраняя большую часть атмосферных и часовых ошибок, тем самым

повышая точность определения местоположения примерно до трех метров. На практике точность WAAS часто не превышает 1,5 метра. Часто этой точности более чем достаточно для большинства обычных GPS-приложений, но недостаточно для картографических и геодезических приложений.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Методы определения навигационных параметров подвижных средств с использованием спутниковой радионавигационной системы ГЛОНАСС: дис. – Красноярск СФУ, 2012. – 260 с.

2. Satellite Based Augmentation Systems [сайт]. – URL: https://novatel.com/an-introduction-to-gnss/chapter-5-resolving-errors/satellite-based-augmentation-systems/ (дата обращения: 27.11.2022).

3. What Are Satellite-Based Augmentation Systems (SBAS) [сайт]. – URL: https://wp.stolaf.edu/it/gis-sbas/ (дата обращения: 27.11.2022).

**УДК 004.891.2**

**Редько С.В.**

# ИССЛЕДОВАНИЕ МЕТОДА КОГГЕРА ДЛЯ РЕШЕНИЯ ЗАДАЧ ДИАЛОГОВОЙ СИСТЕМЫ АЛЬТЕРНАТИВ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В работе проводится анализ существующих математических инструментов системного подхода к решению проблем принятия решений и их сравнение с исследуемым методом Коггера.

Главной задачей при принятии решения является выбор варианта, наилучшего для достижения некоторой цели, или ранжирование множества возможных вариантов по степени их влияния на достижение этой цели.

Следующие задачи принятия решений – поиск критериев оценки альтернатив и преодоление многокритериальности, сама задача выбора, а затем и реализации решений. Существует много методов решения проблем, возникающих на стадиях и этапах процесса принятия решений. Все эти методы в виде соответствующего математического аппарата реализованы в специальных информационных системах - системах поддержки принятия решений (СППР).

В настоящее время, распространены диалоговые системы, которые помогают пользователю выбрать наиболее правильную альтернативу для решения той или иной задачи. Сегодня диалоговые системы получили широкое распространение: мы часто можем встретить их в повседневной жизни, когда звоним в банк, обращаемся в службу технической поддержки, указываем маршрут в навигаторе. Диалоговые интерфейсы

стали основой современных персональных помощников, которые помогают выполнять повседневные задачи.

Существующие современные системы наряду с имеющимися достоинствами обладают рядом недостатков, которые не позволяют использовать их напрямую для решения описанных выше проблем. Все это определяет актуальность диалоговых систем альтернатив на основе метода Коггера.

Применение СППР основано на экономической целесообразности и определяется сложностью задач, которые решают с их помощью.

Диалоговые системы альтернатив предназначены для решения заранее определенного набора задач. Для них характерны короткие беседы, по заранее прописанному разработчиками сценарию.

Каждая целеориентированная диалоговая система создаётся и настраивается под конкретную задачу, содержащая большое число правил, созданных вручную. Все это выделяет основную особенность таких систем – наличие сценария.

Существует множество математических инструментов системного подхода к решению проблем принятия решений. Рассмотрим те, которые используются в данный момент для решения задач диалоговых систем и сравним их.

Традиционными подходами к анализу намерения пользователя считаются методы, основанные на заранее подготовленном наборе правил или использующие статистические методы, например, скрытые марковские модели. Данные методы использовались в ранних диалоговых системах и являлись достаточно трудоемкими в поддержке и имели невысокую точность результатов.

Также применяются различные методы машинного обучения для задачи классификации: наивный байесовский классификатор, метод опорных векторов, логистическая регрессия, метод k-ближайших соседей или комбинацию этих методов. По сравнению с традиционными методами машинного обучения, нейросетевые модели требуют больше данных и ресурсов для обучения, но, тем не менее, показывают высокую точность результатов.

На данный момент довольно распространены системы поддержки принятия решений на основе метода анализа иерархий, который разработан американским ученым Томасом Саати. Для увеличения обоснованности в выборе метода при принятии решений преимущественно применять количественные методы экспертных оценок – метод Дельфи и метод анализа иерархий (МАИ), дающие отображение высококачественных экспертных оценок. К МАИ также относится и исследуемый нами метод Коггера.

В методе анализа иерархий выполнена попытка обойти слабости дельфийского метода, одновременно применяя его сильные стороны,

связанные с модернизированием операций по принятию решений и планирования в условиях неопределенности. В отличие от метода Дельфи, в МАИ поддерживаются многочисленное взаимодействие и дискуссии. Следовательно, появляются новые и значимые знания по мере того, как группа обследует предположения, лежащие в основе персональных суждений. Выпадающие из единого русла мнения предусматриваются при вычислениях, но не удаляются.

МАИ, кроме того, перенимает кое-что из метода Дельфи. В частности, применение анкет аналогично тем, что используются в методе Дельфи для пересылки вопросов многим заинтересованным лицам. Превосходство его в том, что в процесс включаются разные заинтересованные лица, чьи мнения иначе бы было нельзя принять к сведению.

В методе Дельфи и в МАИ процесс анализа задачи доводит до совершенства качество суждений, при этом метод иерархического анализа расчленяет суждение на простые составляющие и вследствие этого, гораздо лучше подходит к познавательной манере человека. Практически все исследователи, применяющие на практике, советовали её применение при планировании и моделировании как краткую и несложную операцию, отражающую воззрения участников с очень успешным, удачным итогом или результатом.

Достоинства и недостатки метода Дельфи и МАИ отображены в таблице 1.

Таблица 1 – Достоинства и недостатки метода Дельфи и МАИ

| Название | Достоинства | Недостатки |
|---|---|---|
| Метод Дельфи | 1.Позволяет устранить необходимость выбора альтернатив<br>2.Сохраняются все преимущества мозгового штурма | 1.Требует много времени для получения решения<br>2.Дорогой метод<br>3.Сложный по своей организации |
| Метод анализа иерархий | 1.Принимает во внимание «человеческий фактор»<br>2.Не находится в зависимости от сферы деятельности, в которой решается<br>3.Дает удобные средства учёта экспертной информации<br>4.Прост по своей организации | 1.Нет средств для выяснения правдивости данных<br>2.Дает лишь метод рейтингования альтернатив |

Полезным преимуществом МАИ считается его конкретное математическое обоснование, объяснение и простота вычислительных алгоритмов, что обеспечило исследование и разработку надежных программных средств его реализации. Практически постоянно для

одинаковой проблемы имеется целый диапазон мнений. Отсюда следует, что это позволяет принимать во внимание «человеческий фактор» при подготовке принятия решения.

В МАИ нет средств для выяснения правдивости данных. Если сбор данных проведён с помощью опытных специалистов, рецензентов, и в собранной информации нет важных противоречий, то качество такой информации признается хорошим.

МАИ дает лишь метод рейтингования альтернатив, хотя не имеет внутренних средств для интерпретации рейтингов. Другими словами, человек, принимающий решение, понимания рейтинг вероятных решений, обязан исходя из ситуации, сам сделать вывод. Это следует признать изъяном метода.

Метод дает возможность выбора удобных средств учёта экспертной информации для решения различных задач, отображает естественный ход человеческого мышления и выделяет наиболее единый подход, чем метод логических цепей.

МАИ является замкнутой логической конструкцией, обеспечивающей при помощи несложных правил анализ трудоемких задач во всем их многообразии и приводящей к лучшему варианту [1].

Таким образом, МАИ является сбалансированным методом решения сложной проблемы, он может успешно использоваться для решения разнообразных задач ранжирования и выбора, однако его эффективность проявляется при поиске решения сложных проблем, требующих системного подхода и привлечения большого числа экспертов.

Рассмотрим такие МАИ, как метод Саати и метод Коггера. Основным объектом в рассматриваемых методах является треугольная матрица, называемая здесь матрицей попарных сравнений. Согласно методу Саати по треугольной матрице строится полнозаполненная матрица.

В методе Саати нижняя часть матрицы заполняется для контроля квалификации эксперта. Например, если эксперт определил, что первый параметр в 4 раза больше второго параметра, то второй параметр должен быть в 4 раза меньше первого параметра. В методе Коггера от этой проверки отказались для упрощения диалога с пользователем.

Для нахождения вектора весовых коэффициентов ранжируемых объектов используется не система уравнения где S – треугольная матрица попарных сравнений что облегчает решение задачи [2, 3].

Таким образом, по сравнению с другими методами, относящимся к МАИ, метод Коггера является наиболее рациональным для решения задач диалоговой системы альтернатив как с точки зрения легкости ведения диалога с пользователем, так и простоты по своей организации.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Андрейчиков А.В., Андрейчикова О.Н. Математические, эвристические и интеллектуальные методы системного анализа и синтеза инноваций.: Учебник – М.: Либроком, 2013, 306 с..

2. Черноруцкий И.Г. Методы принятия решения. – СПб.: БХВ-Петербург, 2005. – С. 126-133.

3. Черненький, А.В. Принципы оценки и ранжирования структурных подразделений вуза [Текст] / А.В. Черненький // Технические науки. – 2016. – КубГАУ, №116(02).


**УДК 519.688**

**Саморукова О.Д., Крошилин А.В.**

# ОБЗОР МАТЕМАТИЧЕСКИХ МЕТОДОВ ПРОГНОЗИРОВАНИЯ БЮДЖЕТА ПРЕДПРИЯТИЯ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В статье рассмотрены основные математические методы, используемые при анализе деятельности предприятий, даны их краткие характеристики, сделан вывод о возможности их применения к прогнозированию бюджета предприятий.

Бюджетирование – это система обязательных процедур и правил на всех этапах, начиная с планирования и заканчивая анализом исполнения бюджета, и, конечно, включая промежуточные этапы – учёт, контроль и принятие решений.

Исполнение бюджета – это управление предприятием в течение всего бюджетного периода с целью достижения финансовых и производственных результатов в соответствии с параметрами утверждённого бюджета [1].

С целью оперативного управления бюджетом предприятия используется уточнённый годовой прогноз, который формируется путём последовательного замещения плановых показателей на фактические и актуализации первоначально сформированных данных на оставшиеся периоды. В сформированном прогнозном бюджете целевые показатели могут отличаться от первоначально запланированных. В таком случае компания может попытаться достигнуть утверждённых целевых показателей путём перераспределения бюджета будущих периодов, либо принять решение о их корректировке [2].

Несомненно, в вопросах исполнения бюджета предприятия важную роль играет вопрос корректного формирования уточнённого годового прогноза. На текущий момент, формированием прогноза, как правило, занимается менеджмент предприятия, путём проведения анализа

фактической ситуации и корректировок показателей будущих периодов. Одним из перспективных направлений процесса бюджетирования является разработка алгоритмов автоматического перераспределения бюджета на оставшиеся периоды. Данные алгоритмы способны повысить точность прогнозирования за счёт возможности учёта план-факт анализов за длительные периоды, а также сократить время принятия решений за счет автоматизации расчётов и индикации ключевых позиций.

Прежде чем к приступить к разработке новых алгоритмов, необходимо провести обзор математических методов, которые могут быть использованы в процессе бюджетирования.

Под методом понимается иерархическая совокупность различных приемов, направленных на разработку прогнозов в целом; также это путь, способ достижения цели, который исходит из знания наиболее общих закономерностей. В прогнозирование большое значение имеет выбранный метод, т.е. одна или несколько математических или логических операций, которые направлены на получение конкретного результата при прогнозировании [3].


Рисунок 1 — Классификация методов прогнозирования

**Статистические методы** основаны на обработке количественной информации для выявления закономерностей и взаимосвязей, с целью построения прогнозных моделей.

**Методы аналогий** основаны на выявлении сходств и различий в закономерностях развития процессов.

**Опережающие методы** базируются на комплексном изучении объекта прогнозирования и научно-технической информации, обладающей свойством опережать развитие объекта прогнозирования.

**Интуитивные методы** прогнозирования базируются на интуитивно-логическом мышлении человека. Они используются в тех случаях, когда невозможно учесть влияние многих факторов из-за значительной сложности объекта прогнозирования или объем слишком прост и не требует проведения трудоемких расчетов.

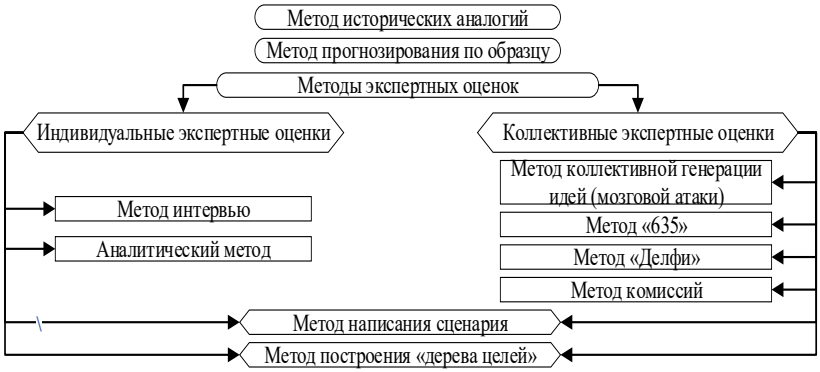


Рисунок 2 — Интуитивные методы прогнозирования

Среди интуитивных методов широкое распространение получили **методы экспертных оценок**. Их сущность заключается в том, что в основу прогноза закладывается мнение специалиста или коллектива специалистов, основанное на профессиональном, научном и практическом опыте.

**Индивидуальные экспертные** оценки основаны на использовании мнений экспертов-специалистов соответствующего профиля. Наиболее широкое распространение получили методы «интервью» и «аналитический».

**Метод «интервью»** предполагает беседу прогнозиста с экспертом по схеме «вопрос-ответ», в процессе которой прогнозист в соответствии с заранее разработанной программой ставит перед экспертом вопросы относительно перспектив развития прогнозируемого объекта.

**Аналитический метод** предусматривает тщательную самостоятельную работу эксперта над анализом тенденций, оценкой состояния и путей развития прогнозируемого объекта. Данный метод мало пригоден для прогнозирования сложных систем из-за ограниченности знаний одного эксперта в смежных областях знаний.

Наиболее достоверными являются **коллективные экспертные оценки.** Методы коллективных экспертных оценок предполагают определение степени согласованности мнений экспертов по перспективным направлениям развития объекта прогнозирования.

В мировой практике широкое применение нашли такие методы коллективных экспертных оценок, как метод коллективной генерации идей, метод «635», метод «Дельфи», метод «комиссий».

Суть **метода коллективной генерации идей (мозговой атаки)** состоит в использовании творческого потенциала специалистов при «мозговой атаке» проблемной ситуации, реализующей вначале генерацию

идей, а затем их деструктурирование (разрушение, критику) с выдвижением – контридей и выработкой согласованной точки зрения

**Метод "635"** — одна из разновидностей «мозговой атаки». Цифры 6, 3, 5 обозначают 6 участников, каждый из которых должен записать 3 идеи в течение 5 мин. Лист ходит по кругу. Таким образом, за полчаса каждый запишет в свой актив 18 идей, а все вместе — 108. Структура идей четко определена. Возможны модификации метода.

Цель **метода "Дельфи"** - разработка программы последовательных многотуровых индивидуальных опросов. Особенности метода "Дельфи":

а) анонимность экспертов. Участники неизвестны друг другу. Взаимодействие членов группы при заполнении анкет исключается;

б) возможность использования результатов предыдущего тура;

в) статистическая характеристика группового мнения.

**Метод "комиссий"** основан на работе специальных комиссий. Группы экспертов за «круглым столом» обсуждают проблему с целью согласования точек зрения и выработки единого мнения.

**Метод написания сценария** относится как к индивидуальным, так и к коллективным экспертным оценкам. Он основан на определении логики процесса во времени при различных условиях и предполагает установление последовательности событий, развивающихся при переходе от существующей ситуации к будущему состоянию объекта.

**Прогнозный сценарий** определяет стратегию развития прогнозируемого объекта. Он обычно носит многовариантный характер и освещает три линии поведения: оптимистическую - развитие системы в наиболее благоприятной ситуации; пессимистическую - развитие системы в наименее благоприятной ситуации; рабочую - развитие системы с учетом противодействия отрицательным факторам, появление которых наиболее вероятно.

**Формализованные методы** базируются на математической теории, которая обеспечивает повышение достоверности и точности прогнозов, сокращает сроки их выполнения и облегчает обработку информации и оценку результатов.


Рисунок 3 — Формализованные методы прогнозирования

**Сущность экстраполяции** заключается в изучении сложившихся в прошлом и настоящем устойчивых тенденций развития объекта прогноза и переносе их на будущее. В ходе данного метода происходит выявление и обоснования тенденций на основе изучения динамических рядов, заключающийся в подборе математических функций, наиболее точно отражающих сложившиеся и прогнозируемые тенденции, а также расчет параметров прогнозных трендов и их графическое построение.

**Формальная экстраполяция** базируется на предположении о сохранении в будущем прошлых и настоящих тенденций развития объекта прогноза; при *прогнозной экстраполяции* фактическое развитие увязывается с гипотезами о динамике исследуемого процесса с учетом изменений влияния различных факторов в перспективе.

В экономическом прогнозировании широко применяется метод математической экстраполяции, означающий распространение закона изменения функции из области её наблюдения на область, лежащую вне отрезка наблюдения.

**Моделирование** предполагает конструирование модели на основе предварительного изучения объекта или процесса, выделения его существенных характеристик или признаков. Прогнозирование экономических процессов с использованием моделей включает разработку модели, её экспериментальный анализ, сопоставление результатов расчетов на основе модели с фактическими данными состояния объекта или процесса, корректировку и уточнение модели.

В современных условиях развитию моделирования и практическому применению моделей стала придаваться особая значимость в связи с усилением роли прогнозирования и переходом к индикативному планированию [4].

Математические методы прогнозирования имеют высокую степень достоверности при анализе деятельности предприятий. Необходимо отметить, что бюджетирование – сложная система, включающая в себя большое количество показателей, как в натуральном, так и в стоимостном выражении, а формирование прогноза исполнения бюджета требует комплексного подхода. Высокая точность прогнозирования может быть достигнута путем определения оптимальных методов прогнозирования для отдельных элементов бюджета, комбинирования описанных выше методов прогнозирования и построения моделей, учитывающих выявленные закономерности и реализующих разработанные алгоритмы с целью формирования прогнозных сценариев исполнения бюджета предприятия.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Бюджетирование. Контроль и анализ бюджета: [Электронный ресурс]. URL: https://alfaseminar.ru/budgeting/3

2. Svetlana Kroshilina, Aleksandr Kroshilin, Aleksandr Pylkin, Valeriia Tishkina, Alexei Evseev Enterprise Management Mobile Assistant based on Using the Theory of Fuzzy Logic and Fuzzy Sets // 2019 1st International Conference on Control Systems, Mathematical Modelling, Automation and Energy Efficiency (SUMMA), pp. 247-249

3. Калачева Н.В., Лазарева Л.В. Методы математического прогнозирования экономического развития предприятий // Научно-методический электронный журнал «Концепт». – 2017. - №1 (январь). – 0,3 п.л. – URL: http://e-koncept.ru/2017/170012.htm

4. Планирование и прогнозирование на предприятии АПК: краткий курс лекций / Сост.: Ю.А. Бутырина // ФГОУ ВО «Саратовский ГАУ». – Саратов, 2016. – 78 с

**УДК 004.4'2**

**Сёмин И.Д.**

# ИССЛЕДОВАНИЕ И РЕАЛИЗАЦИЯ МЕТОДОВ АВТОМАТИЗИРОВАННОГО ФОРМИРОВАНИЯ РЕКОМЕНДАТЕЛЬНОЙ БАЗЫ ПО СОВЕРШЕНСТВОВАНИЮ СПОРТИВНОГО ВОСПИТАНИЯ В ФИЗКУЛЬТУРНО-ОЗДОРОВИТЕЛЬНОМ КОМПЛЕКСЕ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

Обзор отечественных и зарубежных источников в области спортивного воспитания в физкультурно-оздоровительном комплексе.
Формулировка объекта, предмета, цели и задач исследования.
Исследование прикладных и информационных процессов в области формирования автоматизированных рекомендаций в области физкультурно-оздоровительного комплекса.

Система поддержки принятия решений (СППР) – компьютерная автоматизированная система, которая помогает людям, принимающим трудные решения в разных условиях объективно проанализировать предметную деятельность. Для это в СППР необходимо внести входные данные. СППР возникли в результате объединения управленческих информационных систем и систем управления базами данных.

**База знаний** (БЗ) – совокупность моделей, правил и факторов, порождающих анализ и выводы для нахождения решений в сложных задачах некоторой предметной области. БЗ состоит из знаний специалистов разных отраслей жизни.

БЗ имеет значительный объем памяти и специальные средства для хранения в ней знаний. Для БЗ применимы следующие функции:

- Создание, загрузка.
- Актуализация.
- Расширение.
- Обработка и формирование знаний, соответствующих текущей ситуации.

Для выполнения этих функций разрабатываются соответствующие программные средства.

Знания – это хорошо структурированные данные или метаданные.

Существуют следующие модели представления знаний: логические, фреймы, семантические сети, продукционные модели (ПМ).

Рассмотрим подробнее продукционную модель. В настоящее время это самая проработанная и популярная модель представления знаний, особенно в экспертных системах. Продукционная модель состоит из выражений, суть которых заключается в следующем: если выполняется условие – нужно произвести какое-либо действие. Продукционные правила (ПП) содержат специфические знания предметной области о том, какие ещё факторы могут быть учтены, или есть ли специфические данные в БД. ПМ применяются в предметной области, где нет чёткой логики и задачи решаются на основании независимых правил. ПП выражается конструкцией вида: «ЕСЛИ возникает какая-либо ситуация, ТО осуществляется определённый набор действий». Плюсами данной модели являются: наглядность, высокая модульность, простота добавления изменений, лёгкость логического вывода. Недостатком является трудность обеспечения непротиворечивости правил при их большом числе, что требует создания специальных правил для разрешений возникающих в ходе вывода противоречий, и время итогового заключения может быть достаточно большим.

Модель предусматривает разработку ПП, имеющих вид:

Если А1 и А2 и. Аn, то Б1 или Б2 или Бn, где А и Б –некоторые высказывания, к которым применяются логические операции И/ИЛИ. Если высказывания в левой части истинны, то высказывания в правой части тоже истины. Логический вывод в продукционных системах основан на построении прямой и обратной цепочек заключений, образуемых в результате последовательного просмотра левых и правых частей соответствующих правил, вплоть до получения окончательного заключения.

В результате проведённых исследований были изучены материалы соответствующей предметной области, исследованы методы представления информации, база знаний, система поддержки принятия решений.

Подготовлены возможные пути решения проблем, возникающих при работе с большими объёмами данных.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1.Ашимова М. Е., Овечкин Г.В. Применение Big Data при создании программной системы для комплексного анализа метеорологических данных – 2020 г.

2.Крошилин А. В., Крошилина С. В., Тишкина В. В. Использование интеллектуальных методов при разработке системы мониторинга результатов деятельности объектов учёта – 2017 г.

3.Бубнов А. А. Основные характеристики программной системы для оценки надежности программного обеспечения – 2017 г.

**УДК 004.02**

**Серов А.П.**

# ЗАДАЧА АВТОМАТИЗАЦИИ РАБОТЫ КОМПАНИИ СЕРВИСНОГО ОБСЛУЖИВАНИЯ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В статье рассматриваются проблемы в области сервисного обслуживания, постановка задачи графика составления работ, реализация на платформе "1С: Предприятие 8.3".

Сервисное обслуживание – совокупность видов деятельности компании, направленных на поддержание правильной работы приобретённой клиентом техники или услуги. Примерами сервисного обслуживания являются: настройка торговых автоматов, поверка весового оборудования, техническое обслуживание лифтов.

Невыполнение сервисного обслуживания может привести к остановке бизнеса организации, штрафам для сервисных компаний и к человеческим жертвам.

В настоящее время распространены случаи, когда компании, занимающиеся сервисным обслуживанием, используют устаревшие средства автоматизации. 30% компаний использует в качестве системы автоматизации выездного сервиса Excel/email/мессенджеры. 35,7% используют неподходящие для решения задач сервисного обслуживания системы класса CRM [1].

Именно поэтому задача разработки программного обеспечения, автоматизирующего работу компании сервисного обслуживания, является актуальной и практически значимой.

В исследуемой предметной области можно выделить следующие проблемы.

1. Человеческий фактор оператора при обработке заявки, поступившей на почту или по телефону. Диспетчер при создании заявки вручную после разговора с клиентом, может исказить информацию или вспомнить её не в полном объёме, а заявка на почте может быть не замечена. Проблема может быть решена, если клиенты будут сами заполнять заявки на ремонт на предоставленных в приложении формах.

2. Отсутствие учёта запчастей и расходных материалов, заявок и заказ-нарядов, либо учёт этих данных ведётся в разных несвязанных системах. Если заявки, заказ-наряды ведутся в одной системе, а учёт в другой, то при переносе вручную большого количества данных информация может быть перенесена некорректно или не полностью. Необходимо разработать такое приложение, чтобы в нем было обеспечено хранение вышеуказанной информации.

3. Составление графика работ вручную. При отсутствии автоматизированного составления расписания сервису тяжело планировать работу наперёд. При неправильном планировании сроки регламентных работ по сервисным договорам постоянно срываются, клиенты покидают компанию. Данная статья посвящена решению этой проблемы.

В рассматриваемой предметной области существуют объекты, для которых нужно выполнить работы – клиенты компании сервисного обслуживания. У них существует определённое количество единиц оборудования, для которых надо провести плановое обслуживание. Для каждой такой заявки хранится примерное время обслуживания.

Задача состоит в следующем: необходимо оптимально распределить работников по заявкам, чтобы их рабочие дни были полностью загружены заявками.

В компании сервисного обслуживания существует определённое количество работников. В неделе есть ограничение по количеству рабочих дней, а в день ограничен продолжительностью рабочего времени. Для заявки ограничение является её примерным временем выполнения (временем обслуживания единицы техники).

Заявки выполняются работниками независимо друг от друга. Время выполнения заявок меньше продолжительности рабочего дня, причём одну заявку нельзя разделить на несколько дней, так как в таком случае расписание могло бы быть составлено таким образом, что одна задача могла быть разделена между несколькими работниками в течение одного дня.

Одним из возможных способов решения задачи составления графика работ является задача «Упаковка предметов в контейнеры» [2].

Имеется конечное множество объектов $U=\{u_1,u_2,...u_n\}$, размеры (веса) $w(u)$ которых заданы рациональными числами. Требуется упаковать объекты в контейнеры размера $V$, чтобы сумма размеров объектов в

каждом контейнере не превосходила $V$, и чтобы число контейнеров было минимальным.

Применимо к области сервисного обслуживания множество объектов $U$ – это множество задач, контейнеры – рабочие дни, $V$ – продолжительность рабочего дня.

Эвристическим алгоритмом, применяющимся для решения задачи упаковки является алгоритм First Fit Decreasing – FFD (первый подходящий по убыванию).

На вход алгоритма поступает список объектов, упорядоченных по убыванию их размеров. Очередной объект, выбираемый из списка объектов, размещается в первый подходящий контейнер, в котором достаточно свободного места. Если же предмет невозможно разместить ни в один из частично заполненных контейнеров, то добавляется новый контейнер и в него размещается данный объект.

FFD позволяет быстро составить расписание, например, в отличие от метода полного перебора. В выбранной предметной области может возникнуть ситуация, когда нужно быстро пересоставить расписание, начиная с определённого дня. Например, если заявка будет выполняться больше, чем её примерное время выполнения или если один из работников заболеет.

Реализация данной задачи была выполнена на платформе «1С:Предприятие 8.3». Создана конфигурация со справочниками «Работники» (хранит номер работника и ФИО), «Заявки» (номер заявки и время выполнения) и регистром сведений «Занятость работников». Также была разработана обработка «Составление расписания». В коде модуля формы этой обработки используется алгоритм FFD для распределения списка заявок по минимальному количеству дней.

Результатом работы обработки «Составление расписания» является добавленные в регистр сведений «Занятость работников» записи и отчёт сформированного расписания. При составлении расписания подсчитывается процент заполненности всех дней расписания.

Для демонстрации работы алгоритма в базу были добавлены 4 тестовые записи в справочник «Работники». Справочник «Заявки» был заполнен 35 тестовыми записями, продолжительность задачи задавалась случайным образом от 1 до 8 часов включительно.

На форму обработки «Составление расписания» были введены данные представленные в таблице 1.

Таблица 3 – введённые данные на форму обработки «Составление расписания»

| | |
|---|---|
| Продолжительность рабочего дня | 8 часов |
| Количество рабочих дней в неделе | 5 дней |
| Время начала рабочего дня | 8:00 |
| Дата начала составления расписания | 05.12.2022 |

Для введённых данных из таблицы 1 было составлено расписание, хранящееся в регистре сведений «Занятость работников». Отчёт сформированный по этим записям представлен на рисунке 1.

На рисунке 1 приведено расписание только для 2 работников. Для данной выборки процент заполненности дней в расписании составил 96,43%, что позволяет сделать вывод о том, что алгоритм FFD для использованной выборки составляет расписание, в котором практически все рабочие дни полностью заполнены заявками.

| Работник | Время начала | Время конца | Заявка |
|---|---|---|---|
| Васильков В.В. | 08.12.2022 8:00:00 | 08.12.2022 14:00:00 | Заявка 23 |
| Васильков В.В. | 12.12.2022 12:00:00 | 12.12.2022 16:00:00 | Заявка 2 |
| Васильков В.В. | 07.12.2022 15:00:00 | 07.12.2022 16:00:00 | Заявка 6 |
| Васильков В.В. | 06.12.2022 8:00:00 | 06.12.2022 16:00:00 | Заявка 8 |
| Васильков В.В. | 09.12.2022 13:00:00 | 09.12.2022 16:00:00 | Заявка 22 |
| Васильков В.В. | 05.12.2022 8:00:00 | 05.12.2022 16:00:00 | Заявка 35 |
| Васильков В.В. | 09.12.2022 8:00:00 | 09.12.2022 13:00:00 | Заявка 10 |
| Васильков В.В. | 12.12.2022 8:00:00 | 12.12.2022 12:00:00 | Заявка 3 |
| Васильков В.В. | 07.12.2022 8:00:00 | 07.12.2022 15:00:00 | Заявка 16 |
| Васильков В.В. | 08.12.2022 14:00:00 | 08.12.2022 16:00:00 | Заявка 9 |
| Куселёв О.А. | 06.12.2022 8:00:00 | 06.12.2022 15:00:00 | Заявка 34 |
| Куселёв О.А. | 06.12.2022 15:00:00 | 06.12.2022 16:00:00 | Заявка 31 |
| Куселёв О.А. | 07.12.2022 8:00:00 | 07.12.2022 15:00:00 | Заявка 15 |
| Куселёв О.А. | 08.12.2022 8:00:00 | 08.12.2022 14:00:00 | Заявка 19 |
| Куселёв О.А. | 07.12.2022 15:00:00 | 07.12.2022 16:00:00 | Заявка 4 |
| Куселёв О.А. | 05.12.2022 8:00:00 | 05.12.2022 16:00:00 | Заявка 28 |
| Куселёв О.А. | 09.12.2022 13:00:00 | 09.12.2022 16:00:00 | Заявка 20 |
| Куселёв О.А. | 09.12.2022 8:00:00 | 09.12.2022 13:00:00 | Заявка 5 |

Рисунок 1 — Составленное расписание

Таким образом, рассмотрены проблемы и задачи автоматизации работы компании сервисного обслуживания, сделан вывод о том, что необходимо разработать собственное программное обеспечение. Была описана математическая постановка задачи, которая позволила свести задачу составления графика работ к задаче об упаковке предметов в контейнеры. Была реализована конфигурация на платформе "1С:Предприятие 8.3" с возможностью составления расписания работников.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Федулов К.В. Сервисное обслуживание среди интеграторов спутникового мониторинга и транспорта. Исследование компании Okdesk. [Электронный ресурс]. 2020 – Режим доступа: URL: https://okdesk.ru/uploads/gps-smt_research_2020.pdf (Дата обращения: 18.11.2022).

2. Бугаев Ю.В., Коробова Л.А., Гудков С.В. Методы оптимизации развозки грузов потребителям несколькими транспортными средствами // Вестник ВГУИТ. 2021. Т. 83. № 1. С. 466-472.

**УДК 004.021**

**Сидоров К. А., Крошилин А.В.**

# ИСПОЛЬЗОВАНИЕ ИНТЕРПОЛЯЦИИ ХАРАКТЕРИСТИЧЕСКОГО ПОЛИНОМА ДЛЯ МИНИМИЗАЦИИ ПЕРЕДАВАЕМОЙ ИНФОРМАЦИИ ПРИ СИНХРОНИЗАЦИИ ХРАНИЛИЩ ДАННЫХ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В статье рассматривается работа алгоритма минимизации передаваемой информации при синхронизации хранилищ данных – CPISync.

Одним из алгоритмов минимизации передаваемой информации при синхронизации хранилищ данных является алгоритм, основанный на использовании интерполяции характеристических полиномов CPISync (Characteristic Polynomial Interpolation-Based Synchronization). Алгоритм решает проблему уменьшения количества передаваемой информации по сети при выполнении синхронизации двух схожих, но имеющих некоторые различия, наборов данных.

В алгоритме CPISync наборы данных представляются в виде отдельных записей, каждая запись – уникальное целочисленное число. Основная идея CPISync заключается в том, что вся хранимая информация (набор данных) может быть представлена в виде характеристического полинома записей набора данных [2]:

$$\chi_S(Z) = (Z - x_1)(Z - x_2)(Z - x_3)...(Z - x_4), \quad (1)$$

где $x_n$ – уникальная запись информации (целочисленное число).

Пусть имеется два набора данных $S_A$ и $S_B$. Для этих наборов, в соответствии с алгоритмом CPIsync, имеет место следующие равенство [2]

$$\frac{\chi_{S_A}(Z)}{\chi_{S_B}(Z)} = \frac{\chi_{\Delta_A}(Z)}{\chi_{\Delta_B}(Z)}, \quad (2)$$

где $\Delta_A = S_A - S_B$; $\Delta_B = S_B - S_A$.

Выражение (2) показывает, что отношение характеристических полиномов наборов данных $S_A$ и $S_B$ равно отношению разностей множеств $S_A$ и $S_B$.

Рассмотрим работу алгоритма CPISync более подробно по шагам.

1. Компьютеры A и B составляют для синхронизируемых наборов данных $S_A$ и $S_B$ их характеристические полиномы $\chi_{S_A}(z)$ и $\chi_{S_B}(z)$. Затем берутся m точек некоторого конечного поля F и для этих точек вычисляются значения характеристических полиномов [1]. Количество точек m должно быть больше или равно числа различий между наборами данных $S_A$ и $S_B$:

$$m \geq m_A + m_B, \quad (3)$$

где $m_A$ – число различий в множестве $S_A$; $m_B$ – число различий в множестве $S_B$.

Компьютер A отправляет компьютеру B значения характеристического полинома $\chi_{S_A}(z)$, вычисленные для m точек некоторого конечного поля.

2. Компьютер B, для того чтобы найти различия между синхронизируемыми наборами данных, должен интерполировать функцию:

$$R(Z) = \frac{P(Z)}{Q(Z)} = \frac{\chi_{\Delta_A}(Z)}{\chi_{\Delta_B}(Z)}, \quad (4)$$

где $\Delta_A = S_A - S_B$; $\Delta_B = S_B - S_A$.

Интерполяция функции R(Z) выполняется следующим образом. Вычисляются значения функции R(Z) путем деления значений полинома $\chi_{S_A}(z)$ на значения полинома $\chi_{S_B}(z)$, которые были найдены ранее для m точек некоторого конечного поля F. С помощью полученных в результате деления значений выполняется интерполяция функции R(Z). Функция R(Z) является рациональной функцией, поэтому может быть представлена в виде отношения P(Z) и Q(Z) (4). P(Z) и Q(Z) представляют собой многочлены, коэффициенты которых являются отличающимися значения наборов данных $S_A$ и $S_B$. Сумма степеней многочленов P(Z) и Q(Z) равна общему количеству различий между наборами данных $S_A$ и $S_B$.

3. Выполняется синхронизация данных на основе полученных результатов. Коэффициенты многочлена P(Z) отправляются компьютеру A для обновления набора данных $S_A$, а коэффициенты Q(Z) используются для обновления набора данных $S_B$ компьютера B.

Представленное описание алгоритма CPIsync является упрощенным, так как считается, что количество различий между двумя наборами данных (количество точек m) заранее известно. Описание алгоритма с заранее неизвестным количеством различий между данными содержится в диссертации Data synchronization in mobile and distributed networks [2]. На рисунке 1 представлена иллюстрация работы алгоритма CPISync.

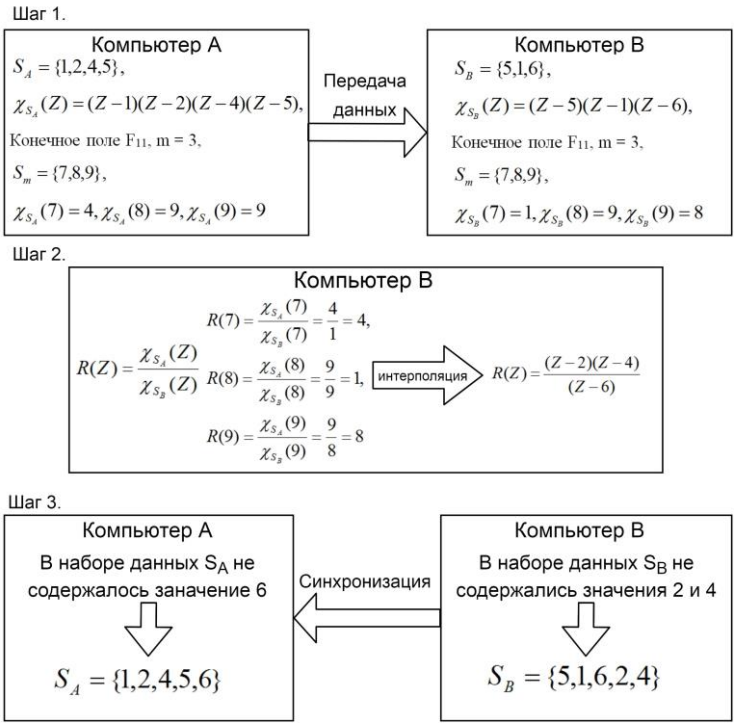


Рисунок 1 - Иллюстрация работы алгоритма CPISync

Преимуществами алгоритма CPISync являются [3] независимость скорости работы алгоритма от объёма синхронизируемых данных (масштабируемость); двусторонняя синхронизация наборов данных.

Недостатками алгоритма CPISync являются [3] зависимость скорости работы алгоритма от количества различий в синхронизируемых наборах данных (чем больше различий в данных, тем меньше скорость работы); неэффективность работы при малых объёмах информации и большом количестве различий; сложность самого алгоритма и его реализации.

Реализация алгоритма CPISync на языке C++ представлена в репозитории «cpisync» веб-сервиса GitHub [4].

Таким образом, алгоритм CPISync позволяет минимизировать количество передаваемой информации по сети за счёт передачи только различающихся частей синхронизируемых данных. Определение отличающихся частей осуществляется благодаря представлению информации в виде характеристических полиномов и последующем использовании их математических свойств. Алгоритм не лишён недостатков, однако при решении определённых видов задач может быть

очень эффективным – позволит сократить время синхронизации большого объёма данных и снизить нагрузку на сеть.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Лидл Р., Нидеррайтер Г. Конечные поля: В 2-х т. Т. 1. Пер. с англ. – М.: Мир, 1988. – 428 с.

2. Agarwal S. K. Data Synchronization in Mobile and Distributed Networks: дис. – Boston University, 2002.

3. Marczewski M. Architecture for multi-party synchronization of data sets in a distributed environment : дис. – Trinity College Dublin. Department of Computer Science, 2005.

4. A library for synchronizing remote data with minimum communication [сайт]. – URL: https://github.com/trachten/cpisync (дата обращения: 20.11.2022).

**УДК 004.02**

**Торжкова А.О.**

# ПРОГНОЗИРОВАНИЕ ПРОДАЖ С ПОМОЩЬЮ АНАЛИЗА ВРЕМЕННЫХ РЯДОВ МЕТОДОМ ЭКСТРАПОЛЯЦИИ ПО СКОЛЬЗЯЩЕЙ СРЕДНЕЙ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В статье рассмотрена реализация прогнозирования с помощью анализа временных рядов методом экстраполяции по скользящей средней и проведена оценка качества прогноза.

Прогнозирование продаж – это обоснованное суждение о величине продаж за некоторый неизвестный период времени в будущем, которое основано на анализе данных ситуации на рынке, предыдущих объёмов продаж, факторов, влияющих на них и т.д.

Для решения задачи прогнозирования продаж необходимо осуществить следующие действия.

1. Получить исходную выборку данных – статистику продаж за определённый период в прошлом, экономические показатели ситуации на рынке и т.д.

2. Выявить факторы, влияющие на количество продаж.

3. Выбрать метод прогнозирования.

4. Применить метод к выборке и получить прогноз.

Для более точного прогнозирования продаж недостаточно учитывать рост и сезонность, необходимо также учесть ещё дополнительные факторы, которые значительно влияют на объём продаж, такие как интенсивность продвижения продукта; ввод новых продуктов; открытие новых направлений продаж; человеческий фактор; производственный фактор; сырьевой фактор и др. Все эти факторы важны, но они резко усложняют любую модель прогноза, и зачастую

пользоваться этими моделями могут только квалифицированные специалисты с экономическим и/или математическим образованием [1].

Для реализации прогнозирования продаж была использована выборка данных продаж телефонов и их аксессуаров за 12 месяцев 2019 года с сайта Kaggle [2]. Выборка хранится в 12 файлах (по месяцам) в формате .csv.

Так как выборка данных была получена из неизвестного источника, то не представляется возможным проанализировать, какие факторы и на сколько влияли на продажи товаров из выборки. Поэтому в дальнейшем влияние факторов будет условно приниматься равным нулю и не учитываться при составлении прогноза.

При анализе различных методов прогнозирования было выявлено следующее. Методы экспертных оценок не могут быть использованы ввиду отсутствия экспертов. Казуальные методы требуют анализа факторов, влияющих на продажи, что не представляется возможным по причинам, описанным выше. Поэтому, а также учитывая большое количество данных в выборке (всего 185 916 записей), для реализации следует выбрать метод анализа временных рядов.

Из множества методов анализа временных рядов был выбран метод экстраполяции по скользящей средней. В основе метода лежит понятие экстраполяции, то есть распространения тенденций временного ряда внутри интервала времени, для которого этот ряд известен, за пределы этого интервала. Данный метод позволяет осуществить краткосрочное прогнозирование временного ряда, сглаживая его путём вычисления средних значений продаж. При этом устраняются случайные колебания величины продаж.

Вычисление прогнозных значений ряда выполняется по следующему алгоритму.

1. Выбирается интервал сглаживания n (количество периодов времени, входящих в интервал).

2. Исходные значения ряда заменяются на средние арифметические внутри выбранного интервала n, где очередной период является серединой интервала. Необходимо отметить, что для нескольких крайних значений ряда вычислить среднее арифметическое будет невозможно.

3. После получения сглаженного временного ряда начинается расчёт прогнозных значений по следующей формуле:

$$y_{t+1} = m_{t-1} + \frac{1}{n} \cdot \left(y_t - y_{t-1}\right), \text{ при n = 3,} \quad (1)$$

где t + 1 – прогнозный период; t – период, предшествующий прогнозному; $y_{t+1}$ – прогнозируемое значение; $m_{t-1}$ – среднее арифметическое значение (скользящая средняя) за два периода до прогнозного; $y_t$ – фактическое значение за период, предшествующий

прогнозному; $y_{t+1}$ – фактическое значение за два периода, предшествующих прогнозному; n – интервал сглаживания.

Для реализации прогнозирования продаж был выбран язык программирования Java с визуализацией исходного графика и графика рассчитанного прогноза на платформе JavaFX.

Все файлы с продажами по месяцам были предварительно объединены в один файл sales2019.csv. Из файла были удалены строки с названиями колонок и строки с заказами за 2020 год, случайно попавшие в выборку. В качестве периода прогноза был выбран месяц.

Программа считывает данные из файлов .csv, преобразует их во временной ряд и производит его экстраполяцию на заданное в тексте программы число периодов. Результат выводится в виде графика при запуске приложения.

На рисунке 1 представлен результат работы программы с заданным количеством прогнозируемых периодов 12.

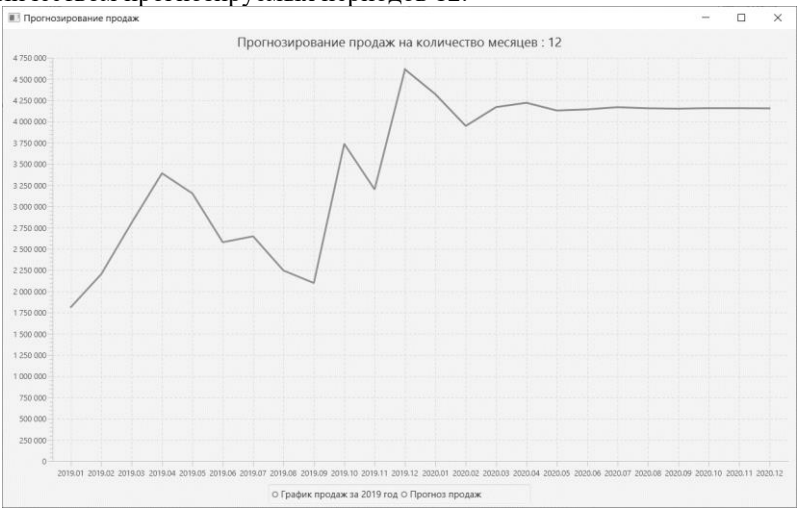


Рисунок 1 – Прогнозирование продаж на 12 месяцев

Оценка качества прогноза.

Из свойств метода скользящей средней очевидно, что оценка качества прогноза путём ретроспективного прогнозирования будет бессмысленной, так как метод не позволяет предсказать колебания продаж, а только определяет общую тенденцию их изменения.

Для оценки качества метода прогнозирования была использована формула расчёта средней относительной ошибки:

$$\varepsilon = \frac{1}{n} \cdot \sum_{i=1}^{n} \left[ \frac{\left| y_ф - y_p \right|}{y_ф} \cdot 100 \right], \quad (2)$$

где n – количество исходных периодов, для которых была рассчитана скользящая средняя, $y_ф$ – фактическое значение величины продаж за период, $y_p$ – рассчитанное значение величины продаж за период (скользящая средняя).

В таблице 1 представлены результаты расчёта средней относительной ошибки.

Таблица 1 – Расчёт средней относительной ошибки

| Период | Продажи, USD | Скользящая средняя, USD | Средняя относительная ошибка, % |
|---|---|---|---|
| Февраль 2019 | 2 202 007,47 | 2 274 230,43 | 3,28 |
| Март 2019 | 2 807 097,39 | 2 799 825,03 | 0,26 |
| Апрель 2019 | 3 390 370,24 | 3 116 641,46 | 8,07 |
| Май 2019 | 3 152 456,75 | 3 040 159,75 | 3,56 |
| Июнь 2019 | 2 577 652,27 | 2 792 578,26 | 8,34 |
| Июль 2019 | 2 647 625,77 | 2 489 881,98 | 5,96 |
| Август 2019 | 2 244 367,89 | 2 329 801,26 | 3,81 |
| Сентябрь 2019 | 2 097 410,13 | 2 692 829,98 | 28,39 |
| Октябрь 2019 | 3 736 711,93 | 3 011 115,09 | 19,42 |
| Ноябрь 2019 | 3 199 223,21 | 3 849 790,27 | 20,34 |
| | | $\varepsilon =$ | 10,14 |

Для данной выборки средняя относительная ошибка составила 10,14%. По таблице 1 также можно заключить, что относительная ошибка тем больше, чем больше колебания временного ряда, то есть метод скользящей средней чувствителен к резким скачкам продаж.

Таким образом, были рассмотрены различные методы прогнозирования продаж, определена проблема выбора метода в зависимости от требований к прогнозированию. При реализации прогнозирования был обоснован выбор метода анализа временных рядов для имеющейся выборки данных, а также рассчитана оценка качества полученного прогноза.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Молчанов Н. Н., Пецольдт К. Выбор метода прогнозирования объема продаж малого предприятия // Экономика и управление. 2019. № 4 (162). С. 51–58.

2. Sales Product Data | Kaggle [Электронный ресурс]. – Режим доступа: https://www.kaggle.com/datasets/knightbearr/sales-product-data, свободный (дата обращения: 18.11.2022)

**УДК 004.4**

**Трифонова Ю.С., Бубнов С.А.**

# ОБОСНОВАНИЕ АКУТАЛЬНОСТИ КОМПЛЕКСНОЙ ОЦЕНКИ СОСТОЯНИЯ ЗДОРОВЬЯ ДЕТЕЙ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В работе рассматривается актуальность комплексной оценки состояния здоровья детей.

Мониторинг здоровья детей относится к одной из актуальных и наиболее значимых проблем здравоохранения, государства и общества. Это определяется тем, что здоровье детского населения является не только интегральным показателем качества здоровья детей и подростков, но и составляет фундаментальную основу для формирования потенциала здоровья взрослых членов общества. Состояние здоровья подрастающего поколения определяет возможности государства по развитию экономики и обеспечению национальной безопасности.

Неблагоприятные демографические процессы в нашем обществе сопровождаются резким ухудшением состояния здоровья детей и подростков. По данным Министерства здравоохранения РФ, заболеваемость детей всех возрастных групп за последние пять лет значительно возросла. Частота болезней костно-мышечной системы увеличилась на 80%, мочеполовой – на 90%, нервной системы и органов чувств – на 35%, системы кровообращения – на 56%, болезней крови и кроветворных органов – на 123%, болезней эндокринной системы – на 90% [1].

Также на актуальность указывают такие факторы, как:

1. стремительный рост числа хронических социально значимых болезней; снижение показателей физического развития; рост психических отклонений и пограничных состояний; рост нарушений в репродуктивной системе;

2. увеличение числа детей, относящихся к группам высокого медико-социального риска;

3. формальный подход к лечению. Зачастую врач после определения диагноза назначает лечебные мероприятия без учёта индивидуальных особенностей пациента;

4. в настоящее время при лечении заболевания врач должен уделять внимание не только физическим причинам возникновения заболевания, но и социальным, а также нервно-психическим возможностям возникновения заболевания.

В основе ухудшения здоровья лежит целый комплекс социально-экономических причин, среди которых можно выделить следующие:

1. несовершенство существующей системы медицинского обследования детей и подростков;
2. ухудшение качества питания;
3. загрязнение окружающей среды;
4. уменьшение объема профилактических программ в амбулаторном звене здравоохранения;
5. рост стрессовых ситуаций в повседневной жизни детей;
6. несовершенство системы психолого-педагогической поддержки школьников и детей дошкольного и раннего возраста из социально неблагополучных семей;
7. отсутствие эффективных образовательных программ, направленных на формирование у детей культуры здоровья и способствующих их гармоничному развитию.

Одна из важнейших биологических особенностей растущего организма заключается в наличии «критических периодов развития», когда диапазон адаптационных реакций ограничивается, а чувствительность организма к экзогенным воздействиям повышается [2].

Термин «критические периоды развития» введен П. Г. Светловым для характеристики тех фаз внутриутробной жизни, когда эмбрион и плод особенно чувствительны к повреждающим экзогенным влияниям, формированию врожденных пороков развития или внутриутробной гипотрофии. Однако критические периоды существуют и в постнатальном развитии ребенка и определяются особым состоянием ЦНС, иммунной системы, обмена веществ и энергии.

В критические периоды организм ребенка оказывается в неустойчивом состоянии, подвергаясь более высокому риску развития пограничных и патологических состояний при воздействии неадекватных его возможностям или патогенных раздражителей (инфекционные агенты, ксенобиотики, токсические радикалы, ионизирующая радиация и др.).

Из всего вышеизложенного, можно сделать вывод о том, что неконтролируемый рост хронической патологии – одна из основных причин ухудшения демографической ситуации в России. Под угрозой оказалось здоровье последующих поколений: у больных родителей рождаются больные дети.

Таким образом, срочные мероприятия по оказанию комплексной оценке состояния здоровья детского здоровья в России просто необходимы, иначе сегодняшние дети окажутся поколением без будущего.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Здравоохранение в России.2021:Стат.сб./Росстат.-М., 2021. – 171с/

2.Ямпольская Ю.А. Физическое развитие и адаптационные возможности современных школьников. // «Российский педиатрический журнал», 1998, № 1, 9-11.


**УДК 004.58**

**Хрустов Н.А.**

# ИССЛЕДОВАНИЕ АЛГОРИТМОВ ВЫБОРА РЕЛЕВАНТНЫХ РЕКОМЕНДАЦИЙ ПО ПОДБОРУ ТАРИФОВ И УСЛУГ В СФЕРЕ ТЕЛЕКОММУНИКАЦИЙ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

Инструменты, которые могут помочь человеку в поиске - рекомендательные системы. Статья о программах и сервисах, которые анализируют интересы пользователей и пытаются предсказать, что именно будет наиболее интересно для конкретного пользователя в данный момент времени

Каждый день в глобальной паутине генерируются и накапливаются огромные объёмы информации. Человеку сложно отобрать интересующую его информацию путём обычного просмотра, так как часто релевантная информация теряется среди больших объёмов данных. В связи с этим, создаются инструменты, которые могут помочь человеку в поиске, предлагая тот контент, который предпочтителен для пользователя. Такие программные средства получили название рекомендательные системы.

Рекомендательные системы – программы и сервисы, которые анализируют интересы пользователей и пытаются предсказать, что именно будет наиболее интересно для конкретного пользователя в данный момент времени [1].

Цель работы – исследование и разработка эффективного алгоритма рекомендательной системы, позволяющей выбирать рекомендации с приемлемым уровнем релевантности в условиях большого числа пользователей при неполной или отсутствующей информации об их предпочтениях.

Для достижения поставленной цели, были выделены следующие задачи:

1. Исследовать виды рекомендательных систем.

2. Спроектировать алгоритм выбора релевантных рекомендаций на основе выбранного подхода.

Существует 4 типа рекомендательных систем [3]:

1. Коллаборативная фильтрация (collaborative filtering).
2. Основанные на контенте (content-based).
3. Основанные на знаниях (knowledge-based).
4. Гибридные (hybrid).

В качестве метода построения рекомендательной системы был выбран алгоритм GroupLens, который активно использует коллаборативную фильтрацию, основанную на пользователях.

Простейший способ построить предсказание нового рейтинга $\hat{r}_{u,i}$ – сумма рейтингов других пользователей, взвешенная их похожестью на пользователя $u$:

$$\hat{r}_{u,i} = \overline{r}_i + \frac{\sum_j (r_{j,i} - \overline{r}_j)\omega_{u,j}}{\sum_j |\omega_{u,j}|} \quad (1)$$

где $u$ всегда будет обозначать пользователей (всего пользователей $N$, $u = 1. . N$). $i$ – предметы (в нашем случае тарифы), которые мы рекомендуем (всего $M$, $i = 1. . M$). $x_u$ – набор (вектор) признаков пользователя, $x_i$ – набор признаков предмета [4].

Следуя формуле (1), задача сводится к подсчету коэффициента $\omega[u, j]$, который описывает «похожесть», или близость пользователей $u$ и $j$. Для этого была используем косинусную меру с учетом обратной частоты [4].

$$\omega_{u,j}^{idf} = \frac{\sum_i f_i^2 r_{u,i} r_{j,i}}{\sqrt{\sum_i (f_i r_{u,i})^2}\sqrt{\sum_i (f_i r_{j,i})^2}}, \quad (2)$$

Для подсчета похожести пользователя $u$ и $j$ требуется вычислить вес для каждого тарифа $i$ [4].

$$f_i = \log \frac{N_i}{N}, \quad (3)$$

где $N$ – общее число пользователей, $N_i$ – число пользователей, оценивших продукт $i$. С помощью этого коэффициента можно добиться, чтобы самые популярные тарифы не рекомендовались всем пользователям, независимо от их предпочтений.

Основной алгоритм выглядит следующим образом:

1. Для всех пользователей вычисляется их средняя оценка.
2. Для всех тарифов вычисляется их вес (согласно формуле (3)) и средняя оценка, поставленная пользователями, оценившими тариф.
3. На основе данных, полученных в пунктах 1 и 2, для каждого тарифа вычисляется ожидаемая оценка текущего пользователя (согласно формулам (1) и (2)).
4. Пользователю выдаются лучшие 3 тарифа, которые ему будут наиболее интересны, согласно ожидаемым оценкам.

Рекомендованные тарифы вычисляются по формуле (1). При этом оценка считается не для каждого тарифа, а только для тех, которые понравились похожим пользователям.

В данной работе были рассмотрены основные виды рекомендательных систем и принципы их построения. Была построена простейшая рекомендационная система, основанная на алгоритме коллаборативной фильтрации пользователей GroupLens. Для определения дистанции между пользователями использовалась косинусная мера с коэффициентом $idf$, смягчающим влияние самых популярных тарифов.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Adomavicius G., Tuzhilin A. Context-aware recommender systems. In Recommender systems handbook. // Springer – 2011.- Pp.:217–253.

2. Methods and metrics for cold-start recommendations. // In Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. ScheinA., PopesculA., UngarL., PennockD . ACM - 2002. – Pp.: 253–260.

3. Jannach D., Zanker M., Felfernig A., Friedrich G. Recommender Systems – An Introduction. Cambridge University Press, 2010. 360. P

4. Николенко С.А. Рекомендательные системы. СПб: Изд-во Центр Речевых Технологий, 2012. 53 с.

**УДК 004.58**

**Цымбалюк О.В.**

# ВИЗУАЛИЗАЦИЯ ДАННЫХ С ПОМОЩЬЮ MATPLOTLIB

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

Визуализация данных и количественная информация с помощью средств библиотеки python matplotlib

Визуализация данных кране важна, когда речь идёт об анализе больших данных, которые хранятся в базах данных, связанных с информационной системой, то есть в зависимости от предметной области данные могут абсолютно разными. Визуальное восприятие намного лучше помогает пользователю воспринимать информацию, полученную от анализа данных.

Обычное визуальное представление количественной информации в схематической форме. К этой группе можно отнести всем известные круговые и линейные диаграммы, гистограммы и спектрограммы, таблицы и различные точечные графики.

Данные при визуализации могут быть преобразованы в форму, усиливающую восприятие и анализ этой информации. Например, карта и полярный график, временная линия и график с параллельными осями, диаграмма Эйлера.

График позволяет выразить идею, которую несут данные, наиболее полно и точно, поэтому очень важно выбрать подходящий тип диаграммы.

Выбор можно осуществить по алгоритму:

1. Определение цели визуализации данных.
2. Определение типа данных.
3. Выбор подходящего графика.

Цели визуализации — это реализация основной идеи информации, это то, ради чего нужно показать выбранные данные, какого эффекта нужно добиться — выявления отношений в информации, показа распределения данных, композиции или сравнения данных

Объектом исследования является современная визуализация данных с помощью средств библиотеки python matplotlib. Предметом исследования являются возможности данной библиотеки, которые актуальны для визуализации результатов, полученных с помощью анализа данных. Целью исследования является нахождение оптимального подхода использования вышеописанной библиотеки для имеющихся задач. Задачами НИР являются:

- Изучение технологии визуализации данных.
- Изучение возможностей библиотеки matplotlib языка python.
- Выявление оптимального подхода к решению определённых задач имеющимися методами.

Прикладной задачей, решаемой с помощью визуализации данных является анализ успеваемости студентов. Так, например, с помощью диаграммы рассеивания можно проследить посещаемость, выполнение учебного плана, тенденции успеваемости как отдельно взятых студентов, так и студенческих групп и факультетов.

Однозначным преимуществом данного решения является возможность более точно выявлять причины неуспеваемости или каких-то недостатков образовательной программы, а не фокусироваться на не имеющих глобального значения показателях.

Так, используя систему ежемесячной бальной аттестации, можно выявить за доли секунды отстающих и успевающих студентов, лишь построив график.

Данный код позволит считать данные из файла и построить диаграмму рассеяния, дальнейший анализ которой значительно упростит работу сотрудников ВУЗа:

```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv('Study.csv')
fig, ax = plt.subplots(figsize=(10, 6))
ax.scatter(x = df['Visit'], y = df['Mark'])
plt.xlabel("Посещаемость")
plt.ylabel("Успеваемость")
plt.show()
```

С помощью проведённого исследования была достигнута цель данной научно-исследовательской работы – визуализации данных посредством библиотеки matplotlib.

Было рассмотрено прикладное решение и его значимость для решаемой задачи.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Никулина Е.А. и Шкляр А.В. «Визуализация данных, как средство принятия решений»

2. Открытый курс машинного обучения. Тема 2: Визуализация данных с Python https://habr.com/ru/company/ods/blog/323210/

**УДК 519.812.3**

**Шилкина М.В., Белов В.В.**

# ПРОЕКТ ДОПОЛНИТЕЛЬНОГО МОДУЛЯ 1С:ЗУП ДЛЯ РАСЧЁТА ВЫРАБОТКИ И ЗАРАБОТНОЙ ПЛАТЫ ОСНОВНЫХ ПРОИЗВОДСТВЕННЫХ РАБОЧИХ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В работе рассматриваются вопросы разработки программного обеспечения для автоматического расчёта выработки и заработной платы основных производственных рабочих на примере производственного предприятия ОА «Елатомский приборный завод».

В настоящей статье излагаются основные сведения по результатам выполнения работ фазы анализа проекта по созданию дополнительного функционального модуля для производственного предприятия. В качестве объекта исследования рассматривается автоматизированная система производственного предприятия Акционерное общество Елатомский приборный завод (АО ЕПЗ). Предметом исследования являются процессы учёта и расчёта выработки и заработной платы основных производственных работников.

К настоящему времени проведено исследование функциональности производственного блока предприятия, разработаны требования к разработке программного обеспечения, разработан алгоритм для расчёта выработки и заработной платы основных производственных рабочих на базе действующего производственного учёта.

**Функциональный аспект модели AS-IS**

Акционерное общество Елатомский приборный завод (далее в тексте АО ЕПЗ). АО ЕПЗ специализируется на производстве физиотерапевтических приборов и медицинских товаров для лечебно-профилактических учреждений. Линейка продукции составляет более 200

единиц, численность работников предприятия 1100 человек, из них 350 рабочих.

**Информационный аспект модели AS-IS**

На предприятии АО ЕПЗ для всех выполняемых работ установлены пооперационные нормы времени.

Действует шестиразрядная система тарификации работ. Каждому разряду соответствует тарифная ставка в рублях. Для выполнения каждого вида работ в технологическом процессе на производство изделия указано рабочий какой профессии будет эту операцию выполнять. Поэтому тарифные ставки установлены для каждой профессии.

Для расчета заработной платы по факту выполнения работ рассчитаны сдельные расценки как произведения тарифной ставки соответствующего разряда и профессии на норму времени на выполнении конкретной операции.

**Структурно-архитектурный аспект модели AS-IS**

Проведённое обследование предприятия выявило, что производство линейки продукции имеет полный производственный цикл от изготовления деталей и сборочных единиц до сборки, приемо-сдаточных испытаний, упаковки изделий и передачи их на хранение на склады готовой продукции. Производство осуществляется в сборочно-монтажном, литьевом, мебельном цехах и цехе производству вакуумных систем для забора крови. В цехах имеются участки, на которых осуществляется производственный цикл по производству деталей и сборочных единиц согласно технологической и конструкторской документации.

Все вышеуказанные действия, а именно планирование, запуск в работу, выпуск, перемещение между участками, перемещение на промежуточные склады и передача на склад проводятся в автоматизированной системе разработанной и действующей на предприятии на базе 1С:Предприятие по установленному алгоритму расчётов и документооборота. На этой же базе планируется разработка дополнительного функционального блока – автоматизированной системы учёта выработки и расчёта заработной платы основных производственных рабочих.

**Аспект поведения модели AS-IS**

Производственный процесс строится следующим образом:

1) в конце отчётного месяца производство получает план сдачи готовых изделий на склад;

2) на основе плана сдачи производственно-диспетчерским отделом предприятия разрабатывается оперативный производственный план выпуска деталей и сборочных единиц в разрезе цехов и участков, и график периодичности запуска и выпуска;

3) в процессе производства осуществляется выпуск деталей и сборочных единиц, передача их на промежуточные производственные склады, сборка изделий, проводятся приемосдаточные испытания, упаковка изделий и передача их на склад готовой продукции.

Обоснование необходимости разработки

В современных условиях рынок и конкуренция предъявляют бизнесу жёсткие условия к достоверности информации и оперативности её получения. Возможность онлайн-получения данных о состоянии производства и финансов компании – залог успешного ведения бизнеса и получения максимальной прибыли. Для реализации преимуществ, открываемых компьютеризацией производственной деятельности, необходима автоматизация бизнес-процессов, и не только основных, но и вспомогательных.

Одним из важнейших вспомогательных (обеспечивающих) процессов производственного предприятия является учёт и расчет выработки и заработной платы работников, выбранный предметом исследования. Актуальность выбранной темы исследования обусловлена тем, что в настоящее время в программном обеспечении 1С:ЗУП (1С:Зарплата и управление персоналом) объекта исследования (АО ЕПЗ) отсутствует модуль по расчёту выработки и заработной платы по сдельным расценкам. В настоящее этот расчёт производится вручную в таблицах Excel диспетчерским отделом предприятия. Ежемесячно расчет производится для 350 рабочих, что обусловливает высокий уровень трудоёмкости реализации важного обеспечивающего процесса.

Необходимость разработки дополнительного программного обеспечения продиктована несколькими факторами:

1.Основные производственные рабочие АО ЕПЗ находятся на сдельной системе оплаты труда, то есть получают заработную плату за количество продукции, произведённую за отчётный период.

2. Процесс расчёта сдельной заработной платы является трудоёмким и ограниченным по времени, так как расчеты за выполненный труд производятся по окончанию отчётного месяца до момента выплаты заработной платы.

За столь короткий временной период для расчёта суммы заработной платы по каждому рабочему сдельщику необходимо выполнить следующий набор работ:

1) регистрация количества выпущенной продукции, которое каждый рабочий произвёл за отчётный месяц; примечание: В России для расчёта заработной платы отчётным периодом принят месяц, поэтому учёт труда и расчёт с работниками производится ежемесячно;

2) оформить и предоставить в бухгалтерию предприятия табель учёта рабочего времени; табель учёта рабочего времени – это документ, в

котором учитывается фактически отработанное работником время в течении календарного времени;

3) оформить и предоставить в бухгалтерию предприятия наряд на сдельную работу, содержащий количество произведённой продукции;

4) рассчитать сумму заработной платы каждому рабочему за количество произведённой в течении месяца продукции; расчёт производится в документе «Наряд на сдельную работу», в котором сумма заработной платы рассчитывается как произведение количества продукции по видам на сдельную расценку, установленную за производство каждого вида продукции;

5) рассчитать выработку каждого работника; показатель выработки рассчитывается как произведение количества выпуска продукции на установленную норму времени на её производство, делённое на количество отработанного времени по табелю;

6) оформить пакет документов (табель, наряд), согласовать его по принятой на предприятии системе документооборота и передать на начисление заработной платы в бухгалтерию предприятия.

Заметим, что оперативный расчёт выработки очень важен для учёта и контроля производственных процессов на предприятии, так как данный показатель даёт возможность оценить уровень выполнения и правильность установки норм труда для расчёта сдельных расценок и суммы ежемесячной заработной платы по факту выполнения производственного задания.

Требования к создаваемому программному обеспечению.

В результате разработки автоматизированной системы должны быть созданы следующие продукты:

1) автоматизированная система на основе используемого производственного блока на базе 1С:Предприятие АО ЕПЗ;

2) нормативно-справочная информация;

3) отчёты и печатные формы;

4) спецификации, техническая документация и руководства пользователя.

Укажем документы, которые позволят осуществить автоматизацию расчетов выработки и заработной платы.

Заданием для работника к выполнению работ является документ «Маршрутный лист». Этот документ выпускается согласно графика запуска и выпуска и содержит информацию по наименованию и количеству продукции, которую нужно выпустить, наименованию цеха, участка, профессии, перечню операций, норме трудоёмкости по каждой операции. Все вышеуказанные данные берутся из справочника «Маршруты» на основании действующей технической документации.

Рабочий сдельщик получает «Маршрутный лист» с заданием, выполняет работы, по факту выполнения сдаёт на склад произведённую

продукцию, и предоставляет «Маршрутный лист» в производственно-диспетчерский отдел. Диспетчер проверяет в программе факт сдачи по приходной накладной, оформленной кладовщиком и закрывает по кнопке «Закрыть» маршрутный лист.

По факту закрытия документа «Маршрутный лист», автоматически выпускается документ «Выпуск продукции», который является основанием для констатации факта выпуска и расчета заработной платы рабочему. Рабочий получает на руки документ, подтверждающий факт выпуска, который использует для проверки начисления заработной платы.

По окончании отчетного месяца диспетчеры переносят в таблицы Excel из документов «Выпуск» фактические данные по количеству произведенной продукции по каждому работнику. Для расчета заработной платы составляется наряд на выполненные работы, который содержит данные по количеству выпуска и сдельной расценке. Сдельные расценки также рассчитываются в таблицах Excel, так как данные по тарифным ставкам в программе отсутствуют.

Автоматизацию работ по расчетам предполагается организовать с помощью внешней обработки по разработанному алгоритму.

Разработанный в процессе выполнения научной работы алгоритм позволит проводить расчет выработки и заработной платы автоматически по факту проведения документа «Выпуск».

Для указанного расчёта будет создан справочник: «Ведомость изменения тарифных ставок», в котором в хронологическом порядке будут записаны тарифные ставки в разрезе должностей и разрядов.

Расчет и хранение данных по трудоемкости и сдельным расценкам будет производится в справочнике «Плановая трудоемкость и сдельные расценки», в котором по каждому изделию в разрезе деталей, сборочных единиц, операций будут храниться данные по нормам трудоемкости и сдельной расценке на каждую операцию.

По факту проведения документа «Выпуск» программа будет производить автоматизированный расчет заработной платы по каждому работнику путем умножения данных по количеству произведенной продукции из документов «Выпуск» на сдельную расценку из регистра «Плановая трудоемкость и сдельные расценки».

Для учета рабочего времени и расчета выработки будет разработан документ «Табель». В нем будет производиться фактический учет рабочего времени и подгружаться данные по объему трудоемкости по факту выпуска, рассчитанные как произведения количества из документов «Выпуск» на норму трудоемкости из регистра «Плановая трудоемкость и сдельная расценка». Количество трудоемкости, деленное на табельное время, покажет процент выработки по каждому рабочему и позволит оценить отклонение нормативного значения выработки от фактического, с

целью оценки эффективности работ и построения технологического процесса.

Учет документов по движению товарно-материальных ценностей в проектируемой системе не предполагается.

По факту проведенного исследования и предложений по повышению эффективности работ можно сделать вывод, что предложенная к разработке автоматизированная системы по расчету выработки и заработной платы основных рабочих сдельщиков на АО ЕПЗ позволит повысить скорость и точность расчетов, убрать трудоемкие операции и ошибки, сократить затраты и снизить себестоимость.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Пелькова С.В. Организация и оплата труда за рубежом // Вопросы структуризации экономики. – 2010. – № 2 . С. 379–384.

2. Мачин К.А. Концептуально-методические основы формирования гибкой адаптивной системы оплаты труда на предприятии // Нормирование и оплата труда в промышленности. – 2013. – № 11. – С.38 42.

3. Нестеров В.И. Показатели эффективности деятельности как основа для внедрения новой системы оплаты труда // Заработная плата. Расчеты. Учет. Налоги. – 2019.– №10.– С. 69–73.

4. Пашуто В.П. Организация, нормирование и оплата труда на предприятии: учебно-практическое пособие. – М.: КнОРУС, 2018. –320 с.

**УДК 004.02**

**Шишкин А.И.**

# АВТОМАТИЗАЦИЯ КЛАССИФИКАЦИИ ЗАЯВОК ТЕХНИЧЕСКОЙ ПОДДЕРЖКИ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

В данной работе рассматривается использование алгоритмов машинного обучения для классификации заявок технической поддержки по их содержанию.

За последние годы с увеличение количества онлайн сервисов определяющим фактором при выборе того продукта, которым будет пользоваться клиент, становится служба технической поддержки. Она должна быстро реагировать на вопросы и качественно решать проблемы.

Пользователи оставляют заявки, которые могут относится к той или иной категории. В зависимости от категории у заявки может быть тот или иной исполнитель. Для классификации потока заявок можно выделить диспетчера. Но, во-первых, это стоит денег (зарплата, налоги, аренда офиса). А во-вторых, на классификацию и маршрутизацию заявки будет потрачено время, и заявка будет решена позже.

Исходя из вышесказанного можно сформулировать цель – автоматизировать классификацию заявок технической поддержки для сокращения времени обработки заявок и уменьшения расходов бизнеса на дополнительных сотрудников.

В данной статье исследуется проблема автоматизации классификации заявок, которая будет решаться с использованием машинного обучения.

Формально данная задача описывается отношением между множеством заявок D = {d1, …, dn} и множеством категорий C = {c1, …, cn}. По некоторой целевой формуле F, называемой классификатором, можно отнести документ di к категории ci. Классификатор может выдавать как точное значение {0, 1}, где 0 – означает, что заявка не относится к категории, а значение 1 – показывает на принадлежность к категории. Также классификатор может определять степень подобия документа di к категории ci, в этом случае классификация называется пороговой.

Для решения определённой задачи классификации заявок D по категориям C необходимо подготовить определённую выборку данных, причём одна из них будет обучающей L, а другая тестовой T, которая нужная для определения качества классификатора.

Исходя из вышесказанного можно сделать вывод, что ключевыми факторами для успешного решения задачи автоматизации классификации заявок технической поддержки являются качество и количество данных в обучающей выборке L, а также выбор подходящего алгоритма для классификатора F.

Существует множество готовых алгоритмов для реализации данной задачи [1], которые отличаются значениями по метрикам.

В рамках классификации заявок была выбрана простая метрика: точность – доля объектов в выборке, для которых алгоритм верно определил категорию.

В библиотеке для Python Scikit-learn [2] есть готовые методы подборов параметров по сетке.

В рамках решения задачи было выбрано 2 алгоритма: метод k ближайших соседей и композиция случайных деревьев.

Для начала необходимо собрать, проанализировать и подготовить данные. Для этого будем использовать библиотеку Pandas [3]. В задаче классификации текстов поле у объектов одно — текст. Задать его алгоритму машинного обучения нельзя. Текст необходимо как-то цифровать и формализовать.

В качестве исходных данных у нас есть выгрузка из 8175 заявок в формате xlsx (рисунок 1). Данный файл содержит 4 поля: id – идентификатор для группы заявки по высказыванию; description –

описание заявки, её текст; category – категория, которой принадлежит заявка; intent – намерение пользователя в заявке.

| | A | B | C | D |
|---|---|---|---|---|
| 1 | id | description | category | intent |
| 2 | BM | I have problems with canceling an order | ORDER | cancel_order |
| 3 | BIM | how can I find information about canceling orders? | ORDER | cancel_order |
| 4 | B | I need help with canceling the last order | ORDER | cancel_order |
| 5 | BIP | could you help me cancelling the last order I made? | ORDER | cancel_order |
| 6 | B | problem with cancelling an order I made | ORDER | cancel_order |
| 7 | BI | can you help me canceling my last order? | ORDER | cancel_order |
| 8 | BE | I do not know how to cancel the last order I have made | ORDER | cancel_order |
| 9 | BM | problems with canceling my orders | ORDER | cancel_order |
| 10 | BM | I have problems with cancelling the last order I have made | ORDER | cancel_order |

Рисунок 1 – Структура исходных данных

После формализации данных на основании них будем классифицировать заявки с использованием алгоритмов, выбранных ранее (рисунок 2).

Обученный алгоритм – композиция случайных деревьев показывает наилучший результат 99%, при количестве деревьев 90 и минимальном количестве объектов в узле 5.

Алгоритм k ближайших соседей показывает наилучший результат в 92% при 7 ближайших соседях, определённых по манхеттенскому расстоянию.

Исходя из этих результатов, можно сделать вывод о том, что в качестве алгоритма классификации, в нашем случае, будет выбрана композиция случайных деревьев.


Рисунок 2 – Результаты обучения алгоритмов классификации

По итогу был разработан классификатор заявок технической поддержки, который в дальнейшем необходимо интегрировать в полноценное приложение. Этот классификатор может быть независимым

сервисом в микросервисной архитектуре [4], к которому будет осуществляться запрос, например, по протоколу REST, содержащий в себе текст заявки, а в ответ сервис будет возвращать категорию заявки.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Batura, Tatiana. (2017). Методы автоматической классификации текстов. Международный журнал Программные продукты и системы. 23. 85-99. 10.15827/0236-235X.117.085-099.

2. Scikit-learn [Электронный ресурс] / Режим доступа: https://scikit-learn.org/stable/

3.Pandas [Электронный ресурс] / Режим доступа: https://pandas.pydata.org/

4. Michaelgmccarthy [Электронный ресурс] / Режим доступа: https://www.michaelgmccarthy.com/2021/03/25/rest-apis-in-a-microservices-architecture/

**УДК 004.93**

**Шурыгина О.В., Цуканова Н.И.**

# РАЗРАБОТКА АРХИТЕКТУРЫ НЕЙРОННОЙ СЕТИ С ПОМОЩЬЮ ЭВОЛЮЦИОННЫХ АЛГОРИТМОВ

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В.Ф. Уткина», Рязань

Рассматривается процесс разработки архитектуры нейронной сети с помощью эволюционных алгоритмов. Приводятся результаты экспериментального исследования качества полученных в результате нейронных сетей.

Эволюционные алгоритмы [1]— направление в искусственном интеллекте (раздел эволюционного моделирования), которое использует и моделирует процессы естественного отбора. Его суть в том, что некоторый постоянный алгоритм будет повторяться, но его составляющие открыты для оптимизации. На рисунке 1 показаны этапы эволюционного программирования. Рассмотрим их.

**Инициализация.** Эволюция должна стартовать с начального решения.

**Дубликация.** Создаётся множество копий текущего решения.

**Мутация.** Каждая копия случайным образом мутирует (изменяется), затем обучается.

**Оценка.** Вычисляется оценка полученного нового решения, зависящая от показателей, продемонстрированных сгенерированным существом.

**Отбор.** После оценки существ лучшим из них разрешается репликация, позволяющая стать основой следующего поколения.

**Вывод.** Эволюция — итеративный процесс, на любом этапе её можно остановить, чтобы получить улучшенную (или ту же самую) версию предыдущего поколения.

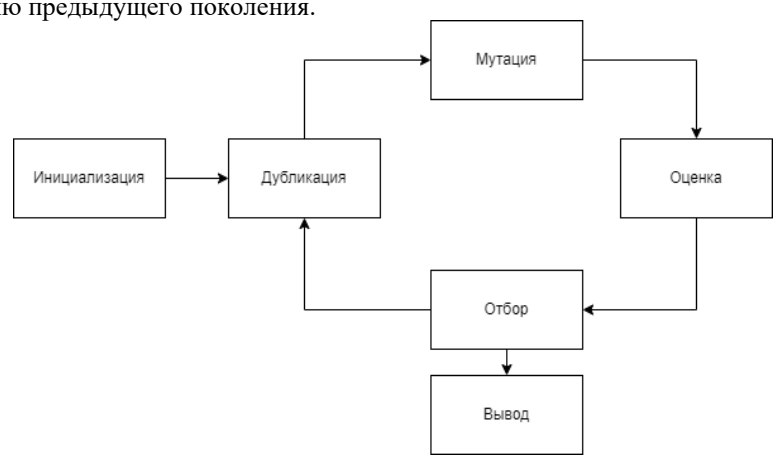


Рисунок 1 – Этапы эволюционного программирования

Воспользуемся эволюционным программированием для поиска лучшей нейронной сети, предназначенной для решения задачи классификации текста. На вход нейронной сети поступает текст произвольной длины и содержания. Необходимо определить, к какому из заданных разделов (или тем) относится этот текст.

На первом этапе следует выбрать структуру модели нейронной сети по следующим правилам. Основой модели будет слой LSTM. В модели всегда должен быть один LSTM слой. Также модель может содержать любое количество слоёв Dense и Dropout. Первым слоем всегда идёт слой embedding, последним – слой Dense. В модели могут меняться количество и вид слоёв, функции активации, оптимизации и потери, размер мини-пакета (batch size) и процент валидационной выборки.

На основе представленного выше описания в программе создан класс модели и вспомогательные классы. Для функций активации, оптимизации, потерь и для типа слоя созданы перечисления. Полями класса модели нейронной сети являются все изменяемые параметры, описанные выше. Количество и вид слоёв хранится в виде массива.

Процесс проектирования нейронной сети можно описать как цикл, состоящий из следующих повторяющихся шагов.

На этапе **Инициализация создаётся** несколько случайных нейронных сетей в соответствии с вышеприведённым описанием. При генерации массива слоёв, полученный массив имеет ровно один слой LSTM. Слои Dense и Dropout генерируются в случайном порядке и количестве. Функции активации, оптимизации и потерь выбираются

случайным образом из перечисления. Размеру мини-пакета (batch size) присваивается случайное число от 50 до 400; проценту валидационной выборки от 0,05 до 0,3.

**Этапы мутации и дупликации.** Основой эволюционного программирования является мутация. При мутации слоёв возможно удаление, добавление или замена слоя. При этом не допускается удаление слоя LSTM или добавление ещё одного. Способ, которым будет производиться мутация слоёв, выбирается случайным образом. Мутация функций активации, оптимизации и потерь представляет собой выбор нового значения из перечисления. Мутация batch size это добавление или удаление из текущего значения случайного числа от 10 до 50. При этом не допускается, чтобы значение batch size выходило из диапазона от 50 до 400. Мутация процента валидационной выборки представляет собой добавление или удаления из текущего значения случайного числа от 0,01 до 0,05. При этом не допускается, чтобы значение выходило из диапазона от 0,05 до 0,3.

Процесс мутации всей модели происходит следующим образом. Случайным образом выбирается один из вышеперечисленных параметров, который затем мутирует.

На этапе **Оценка** после обучения моделей **к**ачество нейронных сетей определяется точностью (accuracy) их работы на тестовой выборке.

На этапе **Отбор** проверяются полученные точности. 50 % нейронных сетей с худшими показателями удаляются. Для каждой из остальных создаются две мутации. Все они будут составлять новое поколение, с которым производится дальнейшая работа.

Эти шаги повторяются заданное количество итераций.

В таблице 1 представлены результаты выполнения программы при следующих начальных параметрах: 20 случайных начальных моделей; 8 итераций; 10 эпох обучения для каждой модели.

Таблица 1 – оценка точности нейронных сетей на каждой итерации

| Итерация | Лучший результат | Средний результат | Худший результат |
|---|---|---|---|
| 1 | 91.08 | 75.247 | 0.66 |
| 2 | 91.43 | 87.88 | 68.90 |
| 3 | 91.95 | 89.566 | 82.9 |
| 4 | 91.78 | 90.23 | 84.42 |
| 5 | 91.958 | 86.36 | 8.01 |
| 6 | 92.18 | 87.29 | 12.01 |
| 7 | 92.07 | 90.07 | 68.32 |
| 8 | 91.94 | 91.22 | 89.16 |

Лучшая точность модели держится на одном уровне. Максимальное полученное увеличение точности 1,1%. Средняя точность моделей увеличивается. Худшая точность ведёт себя непредсказуем. Это

объясняется тем, что изменение любого параметра может значительно снизить точность нейронной сети.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Емельянов В. В., Курейчик В. В., Курейчик В. М. Теория и практика эволюционного моделирования. — М.: Физматлит, 2003. — 432 с.

## СОДЕРЖАНИЕ

**МАТЕМАТИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ**

Межвузовский сборник научных трудов

Подписано в печать 30.01.2023. Формат 60x84 1/16.
Бумага офсетная. Печать цифровая.
Гарнитура «Times New Roman».
Усл. печ. л. 7,75.
Тираж 100 экз. Заказ № 5900

Рязанский государственный радиотехнический
университет имени В.Ф. Уткина
390005, г. Рязань, ул. Гагарина, д. 59/1

Отпечатано в типографии Book Jet
390005, г. Рязань, ул. Пушкина, д. 18
Сайт: http:// book jet.ru
Почта: info@ book jet.ru
Тел.: +7(4912) 466-151

ISBN 978-5-7722-0366-8

ISBN 978-5-7722-0366-8