|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ **Информатика и системы управления**

КАФЕДРА **Компьютерные системы и сети**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.04.01 Информатика и вычислительная техника**

МАГИСТЕРСКАЯ ПРОГРАММА **09.04.01/07 Интеллектуальные системы анализа,**

**обработки и интерпретации больших данных**

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

***К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ***

***МАГИСТРА НА ТЕМУ:***

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

Программная система

прогнозирования появления новых

технологических областей

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-42М |  |  | А.М. Панфилкин |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Руководитель |  |  |  | Д.В. Берёзкин |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Нормоконтролер |  |  |  | С.С. Данилюк |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
|  |  |  |  |  |

*2024 г.*

АННОТАЦИЯ

В настоящее время, с развитием технологий и увеличением объема патентной информации, возникает необходимость в системах, способных прогнозировать появление новых технологических областей на основе анализа патентных данных. Важность таких систем обусловлена стремительным ростом числа патентов, что затрудняет их анализ и прогнозирование вручную. В связи с этим возрастает потребность в разработке программных систем для автоматизированного анализа и прогнозирования на основе патентной информации.

Настоящая выпускная квалификационная работа посвящена разработке программной системы прогнозирования появления новых технологических областей. В рамках работы проводится анализ существующих методов обработки патентных данных, разработка собственного метода построения и оптимизации моделей прогнозирования, проектирование программной системы, реализующей предлагаемый подход. Система предназначена для помощи пользователям-экспертам в анализе патентной информации и прогнозировании новых технологических направлений.

Разработка программной системы выполнена в рамках стратегического проекта, направленного на создание интеллектуальных систем поддержки принятия стратегических решений на базе прогнозной аналитики больших данных. Приложение иллюстрировано графическими материалами и документацией, поясняющими процесс проектирования, особенности структуры компонентов и принципы работы системы.

Работа выполнена в соответствии с программой развития «Приоритет 2030» и входит в перечень результатов проекта «Создание интеллектуальных систем поддержки принятия стратегических решений на базе глубокой и прогнозной аналитики больших данных» [1].

ABSTRACT

Currently, with the development of technology and an increase in the volume of patent information, there is a need for systems capable of predicting the emergence of new technological areas based on the analysis of patent data. The importance of such systems is due to the rapid growth in the number of patents, which makes it difficult to analyze and predict them manually. In this regard, there is an increasing need to develop software systems for automated analysis and forecasting based on patent information.

This final qualifying work is devoted to the development of a software system for forecasting the emergence of new technological fields. As part of the work, an analysis of existing methods of processing patent data is carried out, the development of a proprietary method for constructing and optimizing forecasting models, and the design of a software system implementing the proposed approach. The system is designed to help expert users analyze patent information and predict new technological trends.

The software system was developed as part of a strategic project aimed at creating intelligent systems to support strategic decision-making based on predictive analytics of big data. The application is illustrated with graphic materials and documentation explaining the design process, the features of the component structure and the principles of the system.

The work was carried out in accordance with the development program «Priority 2030» and is included in the list of results of the project «Creation of intelligent strategic decision support systems based on deep and predictive analytics of big data» [1].

РЕФЕРАТ

Расчетно-пояснительная записка 165 стр. (без приложения 122 стр.), 30 рис., 13 табл., 70 источников, 3 прил.

Ключевые слова: ПРОГНОЗИРОВАНИЕ ТЕХНОЛОГИЧЕСКИХ ОБЛАСТЕЙ, АНАЛИЗ ПАТЕНТНОЙ ИНФОРМАЦИИ, КЛАСТЕРИЗАЦИЯ, UMAP, LDA, DBSCAN, ADJUSTED RAND INDEX, ПАТЕНТНЫЙ ЛАНДШАФТ, ИНТЕЛЛЕКТУАЛЬНЫЙ АНАЛИЗ ДАННЫХ

Объектом разработки является программная система прогнозирования появления новых технологических областей на основе анализа патентной информации.

Целью выпускной квалификационной работы является создание программного продукта, который используется для анализа и прогнозирования новых технологических направлений на основе патентных данных. Система предназначена для помощи пользователям-экспертам в идентификации и анализе патентов, а также в прогнозировании появления новых технологических областей.

В процессе выполнения выпускной квалификационной работы проведен анализ существующих методов обработки патентной информации, определены аспекты их использования и области применения. Рассмотрены традиционные подходы к кластеризации и прогнозированию, а также современные методы анализа данных с использованием технологий машинного обучения. В рамках работы были исследованы и применены методы преобразования текстовых данных в векторное представление с использованием модели SentenceTransformer, кластеризация методом DBSCAN и визуализация данных с помощью UMAP и LDA.

В расчетно-пояснительной записке приведены результаты анализа предметной области, описан процесс проектирования структуры компонентов системы на основании технических требований к разрабатываемому программному продукту, выбраны средства разработки. Разработана технология использования приложения пользователем для выполнения анализа и прогнозирования на основе патентных данных. Приведены результаты функционального тестирования программного продукта.

Результатом работы является реализованное с использованием языка программирования Python и специализированных библиотек программное приложение. Разработанная система позволяет пользователю-эксперту загрузить патентные документы в формате .csv, провести их предварительную обработку и анализ, создать модели для прогнозирования на основе кластеризации патентов и визуализировать патентный ландшафт. Система также предоставляет возможности для анализа матрицы ассоциаций, что позволяет формулировать гипотезы о возможном появлении новых технологических областей и их вероятностях.

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 13](#_Toc167826473)

[1 Анализ подходов к прогнозированию появления новых технологий на основе патентной информации 15](#_Toc167826474)

[1.1 Постановка задачи 15](#_Toc167826475)

[1.2 Подход на основе кластеризации 17](#_Toc167826476)

[1.2.1 Концепция 17](#_Toc167826477)

[1.2.2 Существующие решения 20](#_Toc167826478)

[1.2.3 Сравнительный анализ 22](#_Toc167826479)

[1.3 Подход на основе ресурсных сетей 23](#_Toc167826480)

[1.3.1 Концепция 23](#_Toc167826481)

[1.3.2 Существующие решения 24](#_Toc167826482)

[1.3.3 Сравнительный анализ 26](#_Toc167826483)

[1.4 Подход на основе машинного обучения 27](#_Toc167826484)

[1.4.1 Концепция 27](#_Toc167826485)

[1.4.2 Существующие решения 29](#_Toc167826486)

[1.4.3 Сравнительный анализ 48](#_Toc167826487)

[1.5 Методы векторизации текста патентных документов 50](#_Toc167826488)

[1.5.1 TF-IDF 50](#_Toc167826489)

[1.5.2 Bag of Words 52](#_Toc167826490)

[1.5.3 Word Embeddings 52](#_Toc167826491)

[1.5.4 Doc2Vec 53](#_Toc167826492)

[1.5.5 Sentence Transformers 54](#_Toc167826493)

[1.5.6 Сравнение методов векторизации текста патентов 55](#_Toc167826494)

[Выводы 55](#_Toc167826495)

[2 Проектирование и реализация программного продукта 59](#_Toc167826496)

[2.1 Разработка архитектуры программной системы 59](#_Toc167826497)

[2.1.1 Анализ вариантов построения систем 59](#_Toc167826498)

[2.1.2 Структурная схема программной системы 67](#_Toc167826499)

[2.2 Выбор средств разработки 69](#_Toc167826500)

[2.2.1 Машинное обучение 70](#_Toc167826501)

[2.2.2 Визуализация 71](#_Toc167826502)

[2.2.3 Хранение данных 72](#_Toc167826503)

[2.2.4 Клиент-серверное взаимодействие 73](#_Toc167826504)

[2.2.5 Инфраструктура 75](#_Toc167826505)

[2.3 Разработка метода прогнозирования 76](#_Toc167826506)

[2.3.1 Метод построения модели 76](#_Toc167826507)

[2.3.2 Метод построения прогнозной модели 79](#_Toc167826508)

[2.3.3 Метод генерации отчета 82](#_Toc167826509)

[2.4 Разработка концептуальной модели предметной области 85](#_Toc167826510)

[2.5. Разработка программного интерфейса 87](#_Toc167826511)

[2.6. Разработка пользовательского интерфейса 91](#_Toc167826512)

[2.7. Тестирование программного продукта 93](#_Toc167826513)

[2.7.1 Функциональное тестирование 94](#_Toc167826514)

[2.7.2 Тестирование производительности 95](#_Toc167826515)

[Выводы 95](#_Toc167826516)

[3 Разработка методики тестирования и оценки качества прогнозирования системы 97](#_Toc167826517)

[3.1 Разработка алгоритма тестирования 97](#_Toc167826518)

[3.2 Набор данных 100](#_Toc167826519)

[3.3 Метод оценки результатов тестирования 102](#_Toc167826520)

[3.4 Тестирование системы 103](#_Toc167826521)

[3.4.1 Загрузка документов 103](#_Toc167826522)

[3.4.2 Создание наборов данных 104](#_Toc167826523)

[3.4.4 Построение моделей 105](#_Toc167826524)

[3.4.4 Построение прогнозов 106](#_Toc167826525)

[3.5 Результаты тестирования 107](#_Toc167826526)

[Выводы 109](#_Toc167826527)

[ЗАКЛЮЧЕНИЕ 111](#_Toc167826528)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 113](#_Toc167826529)

[ПРИЛОЖЕНИЕ А 123](#_Toc167826530)

[ПРИЛОЖЕНИЕ Б 124](#_Toc167826531)

[ПРИЛОЖЕНИЕ В 125](#_Toc167826532)

# ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

В настоящей работе применяют следующие определения, обозначения и сокращения:

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) – метод кластеризации данных, основанный на плотности точек, что позволяет автоматически определять число кластеров и выделять шумовые данные.

IPC (International Patent Classification) – международная патентная классификация, система для классификации патентов по различным технологическим областям.

UMAP (Uniform Manifold Approximation and Projection) – метод визуализации данных, который позволяет строить двух- или трехмерные представления высоко размерных данных.

LDA (Latent Dirichlet Allocation) – статистическая модель, используемая для тематического моделирования текстов, которая выявляет скрытые темы в наборе текстовых документов.

Adjusted Rand Index (ARI) – метрика, используемая для оценки качества кластеризации, учитывающая количество правильных и неправильных пар кластеров.

Embedding – «встраивание»; процесс представления слов или иных компонентов языка в векторном пространстве, в контексте может обозначать само векторное представление данных.

SentenceTransformer – модель для создания векторных представлений предложений (sentence embeddings), основанная на BERT.

Python – язык программирования, используемый для разработки программного обеспечения.

Токен – минимальная единица представления и обработки текста; в контексте модели BERT токен представляет собой минимальную единицу текста (слово или часть слова), которая подается на вход модели для обработки, после чего каждому токену будет соответствовать уникальный идентификатор и векторное представление (embedding).

BERT (Bidirectional Encoder Representations from Transformers) – модель-кодировщик глубокого обучения, основанная на архитектуре Transformers, используемая для обработки естественного языка.

.csv (Comma-Separated Values) – формат текстовых файлов, используемый для представления табличных данных, в котором значения отдельных ячеек разделяются запятыми.

Кластеризация – процесс группировки объектов в кластеры на основе схожих характеристик.

Матрица ассоциаций – таблица, отображающая степень соответствия между кластерами базовой и прогнозной моделей.

Машинное обучение (ML) – область искусственного интеллекта, которая разрабатывает алгоритмы для обучения компьютеров на основе данных и улучшения их работы без явного программирования.

Скрытая марковская модель (HMM) – статистическая модель, используемая для анализа последовательностей с скрытыми состояниями.

Сервис-ориентированная архитектура (SOA) – подход к разработке программных систем, при котором функциональность приложения разделяется на автономные сервисы, взаимодействующие через стандартизированные интерфейсы.

Микросервисная архитектура (MSA) – подход к разработке программного обеспечения, при котором приложение строится как набор независимых микросервисов, каждый из которых выполняет определенную функцию.

Pooling – метод уменьшения размерности данных в нейронных сетях, сохраняющий важную информацию.

Среднеквадратичная ошибка (MSE) – функция потерь, используемая для оценки качества модели, измеряя среднюю квадратичную разницу между предсказанными и реальными значениями.

Кросс-энтропия – функция потерь, используемая для оценки качества классификационных моделей, измеряя разницу между распределением предсказанных и реальных классов.

Метод опорных векторов (SVM) – алгоритм машинного обучения, который строит гиперплоскость для разделения классов данных, максимизируя зазор между ними.

Дерево решений – модель машинного обучения, представляющая собой древовидную структуру, где узлы представляют собой проверки на определенные атрибуты, а ветви – результаты этих проверок.

Градиентный бустинг деревьев решений (GBDT) – ансамблевый метод машинного обучения, который последовательно строит модели, каждая из которых корректирует ошибки предыдущей.

Свёрточная нейронная сеть (CNN) – тип нейронной сети, эффективно обрабатывающий данные с сильной пространственной зависимостью, такие как изображения и текст.

Рекуррентная нейронная сеть (RNN) – тип нейронной сети, способный работать с последовательными данными, такими как временные ряды и текст.

# ВВЕДЕНИЕ

В современном мире наблюдается значительное увеличение объемов патентной информации, что связано с интенсивным развитием технологий и глобальной цифровизацией. Ежегодно публикуются тысячи новых патентов, содержащих ценные данные о технических инновациях и тенденциях в различных отраслях. Патенты являются важным источником информации для компаний, исследовательских организаций и государственных учреждений, поскольку они помогают отслеживать развитие технологий и предсказывать будущие направления научных исследований и промышленных разработок. Однако анализ и обработка такого огромного количества патентных данных вручную представляют собой сложную задачу, требующую значительных временных и ресурсных затрат.

Для решения этой проблемы возрастает необходимость в автоматизированных системах анализа патентной информации. Программные системы, способные эффективно обрабатывать и анализировать большие объемы патентных данных, могут существенно сократить время на их обработку и повысить точность прогнозов относительно появления новых технологических областей. Важность разработки таких систем объясняется также потребностью в стратегическом планировании и принятии решений, основанных на достоверной и своевременной информации о технологических тенденциях [2].

В настоящее время существует множество подходов к анализу и обработке патентных данных, включая методы машинного обучения и обработки естественного языка (Natural Language Processing, NLP). В последние годы технологии NLP и глубокого обучения достигли значительных успехов, что открывает новые возможности для создания более совершенных и эффективных систем анализа патентов. Одной из главных задач в этом контексте является прогнозирование появления новых технологических областей, что требует использования сложных алгоритмов кластеризации и классификации данных [3].

В данной работе рассматривается процесс разработки программной системы для прогнозирования появления новых технологических областей на основе патентной информации. Система позволяет загружать и обрабатывать данные из патентных документов, преобразовывать текстовую информацию в векторное представление, проводить кластеризацию патентов и визуализацию патентного ландшафта. Основной акцент делается на оптимизации порога кластеризации и ассоциации кластеров с классами IPC (International Patent Classification) [4], что позволяет улучшить точность прогнозов и формировать обоснованные гипотезы о появлении новых технологий.

Настоящая работа включает анализ существующих методов и технологий, описание разработанного алгоритма и структуры системы, а также результаты экспериментов и тестирования программного продукта. Разработка системы выполнена с использованием языка программирования Python и современных библиотек для обработки данных и машинного обучения. Программный продукт предназначен для использования экспертами в области патентного анализа и может существенно повысить эффективность их работы за счет автоматизации рутинных процессов и предоставления более точных и обоснованных прогнозов.

# 1 Анализ подходов к прогнозированию появления новых технологий на основе патентной информации

## 1.1 Постановка задачи

Для начала формализуем задачу прогнозирования появления новых технологических областей на основе анализа патентной информации. Прогнозирование заключается в предсказании возникновения новых объектов патентного классификатора, который имеет древовидную структуру, включающую секции, классы и группы. Каждый патент, выданный патентными бюро, ассоциируется с одним из объектов классификатора.

Эволюция патентного классификатора может быть представлена следующими элементами:

* – множество объектов патентного классификатора, где – объект классификатора с временем создания ; объект в любой момент времени описывается моделью , отражающей его текущее состояние;
* – множество патентов, где – патент с временем выдачи ; патент в любой момент времени описывается моделью ;
* – множество иерархических связей между объектами классификатора, где связь указывает на то, что объект является дочерним по отношению к , и характеризуется временем возникновения ;
* – множество трансферных связей между объектами классификатора, где связь указывает на трансфер из объекта в , и характеризуется временем возникновения ;
* – множество связей принадлежности патентов объектам классификатора, где связь указывает на то, что патент относится к объекту , и характеризуется временем возникновения ;
* – множество ссылочных связей между патентами, где связь указывает на то, что патент ссылается на патент , и характеризуется временем возникновения .

Эволюцию классификатора можно графически представить в виде графа , где – множество узлов, соответствующих объектам классификатора и патентам, а – множество направленных рёбер, соответствующих различным видам связей между узлами.

При появлении нового объекта классификатора в граф добавляется узел и ребро , ведущее от к его родительскому объекту , а также ребра , ведущие к от объектов , из которых выполнен трансфер в . При появлении нового патента в граф добавляется узел и ребро , ведущее от к объекту , к которому относится этот патент. Также добавляются ссылочные ребра , ведущие от к ранее выданным патентам . Пример графа представлен на рисунке 1.

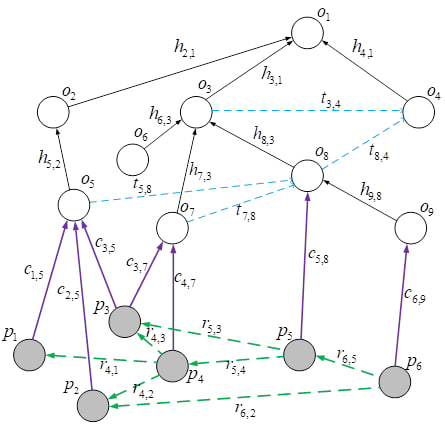


Рисунок 1 – Графовое представление эволюции классификатора

Предполагается, что появление нового объекта можно предсказать на основе анализа развития существующих объектов и характеристик недавно появившихся патентов. Это связано с тем, что новые объекты классификатора обычно отражают новые научные или технологические направления, которые не могут быть учтены в текущем классификаторе. Поэтому, перед появлением нового объекта, появляются патенты, относящиеся к этому направлению и временно присваиваемые существующим объектам классификатора.

Таким образом, задача прогнозирования эволюции патентного классификатора заключается в предсказании появления новых объектов классификатора в ближайшем будущем на основе текущего состояния графа .

Для достижения этой цели будут рассмотрены различные подходы к анализу патентной информации, включая методы обработки естественного языка и машинного обучения. Будет разработана система, позволяющая загружать патентные данные, проводить их анализ, строить модели прогнозирования и визуализировать результаты.

## 1.2 Подход на основе кластеризации

### 1.2.1 Концепция

Подход на основе кластеризации патентов представляет собой методологию, направленную на выявление и прогнозирование новых технологических направлений путем группировки патентов по их схожим характеристикам. Этот подход включает несколько компонентов, каждый из которых играет важную роль в анализе патентных данных.

Первым шагом в этом процессе является преобразование текстовой информации патентов в числовые векторы, которые можно использовать для машинного обучения. Для этого применяются различные техники векторизации:

1. TF-IDF (Term Frequency-Inverse Document Frequency) – этот метод оценивает важность терминов в документе относительно всего корпуса документов, что позволяет выделить ключевые слова и фразы, характерные для конкретного патента [5];
2. Word Embeddings – модели, такие как Word2Vec [6], GloVe [7] и FastText [8], преобразуют слова в векторы, отражающие их семантическое значение и контекст; эти модели могут быть расширены до уровня предложений и документов с помощью Sentence-BERT [9] или Universal Sentence Encoder [10];
3. контекстуальные модели – современные трансформерные модели, такие как BERT [11], GPT-3 [12], могут создавать глубокие контекстуальные представления текста, что позволяет учитывать более сложные взаимосвязи между словами и фразами.

После векторизации текста патентов следующие шаги включают применение различных алгоритмов кластеризации для группировки патентов:

1. K-means [13] – этот алгоритм разбивает данные на кластеров, минимизируя внутрикластерное расстояние; он прост в реализации, но требует предварительного задания числа кластеров;
2. иерархическая кластеризация [14] – построение дендрограммы (рисунок 2), где кластеры формируются путем последовательного объединения или разделения групп данных; это позволяет визуализировать иерархические отношения между патентами;

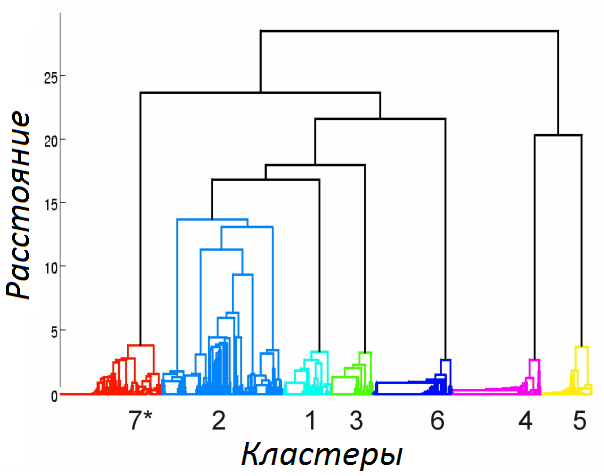


Рисунок 2 – Дендрограмма, отражающая иерархическую структуру кластеров

1. DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [15] – этот метод выделяет кластеры на основе плотности точек в пространстве векторов, что позволяет автоматически определять число кластеров и выделять шумовые данные;
2. Affinity Propagation [16] – алгоритм, который определяет центры кластеров путем передачи сообщений между точками данных, что позволяет эффективно группировать большие объемы данных.

Важным аспектом подхода является анализ изменений в структуре кластеров с течением времени (рисунок 3).



Рисунок 3 – Различные виды эволюции кластеров

Сравнение кластерных структур, полученных на разных временных интервалах, позволяет выявить:

1. рост или сокращение кластеров – изменение размера кластеров может указывать на увеличение или уменьшение интереса к определенным технологическим областям;
2. объединение или разделение кластеров – эти события могут сигнализировать о слиянии технологических направлений или появлении новых, специализированных областей;
3. появление новых кластеров – новые кластеры могут свидетельствовать о зарождении новых технологических направлений.

Эффективная визуализация результатов кластеризации помогает исследователям и аналитикам лучше понимать и интерпретировать сложные взаимосвязи между патентами и технологическими областями. Для этого используются различные инструменты визуализации:

1. дендрограммы – визуализация иерархических кластеров, позволяющая увидеть структуру и связи между кластерами;
2. тепловые карты – показ близости между патентами и кластерами;
3. сетевые графы – отображение связей и взаимодействий между патентами и кластерами.

Эти методы и подходы формируют твердую основу для разработки систем прогнозирования технологических трендов, позволяя не только анализировать текущее состояние научно-технического развития, но и предсказывать возможные направления будущего развития на основе анализа патентных данных. Это имеет важное значение для стратегического планирования и принятия решений в области научно-исследовательской и инновационной деятельности​.

### 1.2.2 Существующие решения

*K-means и K-medoids кластеризация*

Метод K-means является одним из наиболее широко используемых алгоритмов для кластеризации данных. В контексте анализа патентов этот метод позволяет группировать патенты по их схожести, что помогает выявлять основные технологические направления. В исследовании [17] использовался K-means для классификации данных патентов в группы с похожими технологиями. Оптимальное число кластеров определялось и оценивалось с помощью метрик, таких как ширина силуэта, индекс Дэвиса-Боулдина и псевдо F. Этот подход позволил авторам выделить ключевые темы в технологических областях и прогнозировать новые направления развития на основе текстового анализа патентов.

В исследовании [18] использовался метод K-medoids, который схож с K-means, но вместо центроидов использует реальные объекты данных в качестве центров кластеров, что делает его более устойчивым к выбросам. Авторы предложили модель прогнозирования вакантных технологий, используя карту матриц и кластеризацию K-medoids для выявления областей с недостаточной технологической активностью. Результаты показали, что предложенная модель может быть применена в различных технологических областях, включая управление НИОКР, маркетинг технологий и управление интеллектуальной собственностью​​.

*Иерархическая кластеризация и дендрограммы*

Иерархическая кластеризация строит дендрограмму, представляющую иерархическую структуру кластеров. Этот метод позволяет визуализировать отношения между патентами и легко интерпретировать результаты. В исследовании [19] был использован анализ ключевых слов и главных компонент для создания патентной карты, которая помогла выявить пустые области или технологические пробелы. Исследователи использовали иерархическую кластеризацию для построения дендрограммы патентов, что позволило более точно определять ключевые темы и прогнозировать появление новых технологий​​.

*Аффинная пропагация (Affinity Propagation)*

Аффинная пропагация является современным методом кластеризации, который передает сообщения между точками данных для определения центров кластеров. Этот метод особенно эффективен при работе с большими объемами данных, что делает его подходящим для анализа патентов. В работе [20] применили аффинную пропагацию для кластеризации патентов в области беспроводной передачи энергии. Этот метод показал высокую эффективность в выявлении главных технологических направлений и прогнозировании появления новых областей на основе анализа текстов патентов.

*Анализ ключевых слов и тематическое моделирование*

Методы анализа текста, такие как TF-IDF и латентное распределительное моделирование (Latent Dirichlet Allocation, LDA) [21], широко используются для выявления тем в патентных данных. Эти методы позволяют выделять ключевые слова и темы, которые характеризуют группы патентов. В исследовании [22] использовался метод ассоциационного правила майнинга (Association Rule Mining, ARM) [23] для создания кластеров патентов на основе их IPC-кодов и ключевых слов. Этот подход позволил выявлять скрытые связи между патентами и прогнозировать новые технологические направления на основе текстового анализа​​.

### 1.2.3 Сравнительный анализ

Каждый из рассмотренных методов имеет свои уникальные преимущества и недостатки, которые определяют их применение в анализе патентных данных. Сравнительный анализ приведен в таблице 1.

Таблица 1 – Аспекты применения подхода на основе правил

| **Метод** | **Преимущества** | **Недостатки** | **Применение в исследованиях** |
| --- | --- | --- | --- |
| K-means и K-medoids кластеризация | Простота и эффективность, хорошая интерпретируемость, быстрая обработка | Зависимость от начальных условий, требование задания числа кластеров, чувствительность к выбросам | Использовался в исследованиях [17] и [18] для группировки патентов и определения технологических направлений |
| Иерархическая кластеризация и дендрограммы | Иерархическое представление, отсутствие необходимости задавать число кластеров, высокая интерпретируемость | Высокая вычислительная сложность, трудности в масштабировании, сложность интерпретации при больших данных | Применялся в исследовании [19] для создания патентной карты и выявления технологических пробелов |
| Аффинная пропагация (Affinity Propagation) | Автоматическое определение числа кластеров, высокая точность, эффективность для больших данных | Высокие вычислительные затраты, чувствительность к параметрам, сложность интерпретации | Применялся [20] для анализа патентов в области беспроводной передачи энергии |
| Анализ ключевых слов и тематическое моделирование | Выявление тем и ключевых слов, гибкость в анализе, хорошая интерпретируемость | Чувствительность к выбору параметров, ограниченная масштабируемость, сложность в интерпретации тем | Использовался в [22] для создания кластеров патентов на основе IPC-кодов и ключевых слов |

K-means и K-medoids кластеризация являются простыми и эффективными методами, но чувствительны к выбросам и требуют задания числа кластеров. Иерархическая кластеризация предоставляет иерархическое представление данных, но имеет высокую вычислительную сложность. Аффинная пропагация автоматизирует определение числа кластеров и показывает высокую точность, но требует значительных вычислительных ресурсов. Анализ ключевых слов и тематическое моделирование гибки и хорошо интерпретируемы, но зависят от выбора параметров.

Эти методы могут быть использованы в комбинации для достижения наилучших результатов в прогнозировании новых технологических направлений и выявлении вакантных областей в патентных данных.

## 1.3 Подход на основе ресурсных сетей

### 1.3.1 Концепция

Подход на основе ресурсных сетей представляет собой методологию, направленную на прогнозирование появления новых технологических направлений путем моделирования распространения ресурсов (например, патентов) по графу, где вершины представляют собой патентные классы или технологические области, а ребра – связи между ними. Этот подход включает несколько компонентов, каждый из которых играет важную роль в анализе и прогнозировании.

Ресурсная сеть представляет собой ориентированный граф, в котором вершины могут хранить ресурсы (например, патенты или патентные семейства), а ребра обладают определенной пропускной способностью, которая ограничивает объем ресурса, передаваемого между вершинами. На каждом такте дискретного времени происходит перераспределение ресурса между вершинами, соблюдая закон сохранения.

Для моделирования распространения ресурсов используются марковские цепи. Пусть множество возможных состояний системы, функционирующей в дискретном времени. В каждый момент времени система может находиться в одном из состояний . Вероятности перехода из одного состояния в другое задаются стохастической матрицей , где - вероятность перехода из состояния в состояние . Такая система называется цепью Маркова:

Если не зависит от , цепь Маркова называется однородной. Любая стохастическая матрица с вектором начальных вероятностей задает однородную цепь Маркова. Для однородной цепи Маркова с начальным распределением и переходной матрицей выполняется равенство , где – распределение вероятностей в момент времени .

Распределение ресурсов в сети моделируется аналогично марковским цепям. Пусть граф задан матрицей смежности , где – веса дуг. Построим матрицу , элементы которой задают пропорции, в которых разделится ресурс, выходящий из каждой вершины:

где .

Пусть – вектор-строка начального распределения ресурса. Правило распределения ресурса на такте задается аналогично:

где – распределение ресурса на такте .

Ресурсные сети позволяют прогнозировать появление новых объектов патентного классификатора. Для этого моделируется процесс распространения патентов по графу и анализируется достижение пороговых значений ресурса в вершинах. Если определенная вершина получает ресурс, превышающий пороговое значение, это может указывать на возникновение нового технологического направления.

### 1.3.2 Существующие решения

*Скрытые марковские модели (HMM) и Байесовские сети*

Скрытые марковские модели (HMM) в сочетании с Байесовскими сетями представляют собой достаточно хороший инструмент для прогнозирования технологических тенденций. HMM позволяют моделировать динамические системы, где наблюдаемые события зависят от скрытых состояний, не видимых непосредственно. Байесовские сети вносят в эту модель структурированное вероятностное представление взаимосвязей между различными переменными, что позволяет анализировать сложные зависимости и делать обоснованные прогнозы на основе наблюдаемых данных.

В контексте технологий HMM могут использоваться [24] для анализа последовательностей патентов, определения фаз развития технологических направлений и выявления скрытых тенденций, которые могут указывать на новые возможности или риски. Байесовские сети помогают углубить этот анализ, предоставляя инструменты для оценки вероятности различных сценариев развития технологических трендов. Это включает в себя прогнозирование вероятности внедрения новых технологий или появления новых рыночных ниш на основе текущих инновационных активностей и научных исследований.

Такие модели особенно хорошо себя показывают в условиях высокой неопределенности и быстрых изменений в технологических областях [25], поскольку позволяют сформулировать обоснованные прогнозы, опираясь на комплексный анализ больших объемов данных.

*Марковские процессы для системного анализа*

Метод марковских процессов для системного анализа использует стохастические модели для моделирования и прогнозирования динамических систем [26], где изменения состояний зависят только от текущего состояния, а не от истории прошлых состояний. Это особенно полезно для анализа сложных систем, таких как технологические инновации или патентные данные.

В контексте прогнозирования новых технологий, марковские процессы могут помочь в идентификации вероятных будущих технологических трендов на основе текущей активности в патентных данных. Модель может отслеживать, как технологии развиваются или исчезают, предоставляя ценные инсайты для стратегического планирования и развития инноваций.

Преимущества этого подхода включают его способность к моделированию дискретных событий во времени и простоту в применении к различным типам данных. Однако, метод требует точной настройки переходных вероятностей и может быть сложен в интерпретации при анализе сложных или многомерных данных.

*Случайные блуждания и марковские цепи*

Третий метод, связанный с использованием случайных блужданий и марковских цепей для анализа патентных графов, представляет собой подход, который моделирует вероятностные переходы между различными состояниями или патентными категориями в графе. В этом контексте, граф символизирует технологические связи, где вершины представляют различные патентные классы или технологические области, а ребра отражают вероятность перехода от одного состояния к другому.

С помощью марковских цепей, каждое состояние системы (или технологическая область) зависит только от своего предыдущего состояния, что позволяет моделировать эволюцию технологических направлений с течением времени [27]. Случайные блуждания в таких моделях помогают исследовать, как инновации могут переходить между различными областями, основываясь на вероятностных правилах, что может привести к появлению новых технологических трендов и кластеров.

Этот подход особенно полезен для анализа динамических изменений в патентных данных, позволяя идентифицировать потенциальные новые технологические направления на основе текущих тенденций и взаимосвязей.

### 1.3.3 Сравнительный анализ

Сравнительный анализ рассмотренных методов приведен в таблице 2.

Таблица 2 – Сравнительный анализ методов на основе ресурсных сетей

| **Метод** | **Преимущества** | **Недостатки** | **Применение в исследованиях** |
| --- | --- | --- | --- |
| Скрытые марковские модели и Байесовские сети | Моделирование сложных зависимостей, эффективное использование данных для прогнозирования | Требуется значительное количество данных, сложность настройки и интерпретации | [25] |

Продолжение таблицы 2

|  |  |  |  |
| --- | --- | --- | --- |
| **Метод** | **Преимущества** | **Недостатки** | **Применение в исследованиях** |
| Марковские процессы для системного анализа | Простота и мощность модели, возможность моделирования систем с дискретными состояниями | Ограничен при моделировании систем с более сложной зависимостью от истории, требуется точная настройка вероятностей переходов | [26] |
| Анализ Случайные блуждания и марковские цепи | Хорошая адаптивность для моделирования сетевых структур, способность учитывать временную динамику | Высокие вычислительные затраты для больших сетей, необходимость калибровки параметров | [27] |

Рассмотренные методы позволяют глубоко понять динамику развития технологий и могут служить основой для стратегического планирования в инновационной деятельности и исследований. Каждый из этих методов имеет свои сильные стороны в определенных сценариях, а выбор конкретного метода должен основываться на целях исследования, доступных данных и специфических требованиях к анализу.

## 1.4 Подход на основе машинного обучения

### 1.4.1 Концепция

Методы машинного обучения заняли твердую позицию в прогнозировании появления новых технологических направлений на основе патентной информации. Эти методы позволяют выявлять сложные зависимости в данных и делать точные предсказания. В данной главе рассматриваются основные принципы подхода на основе машинного обучения, включая вектор признаков, целевую переменную, методы регрессии и классификации, а также преимущества и особенности применения этих методов в контексте анализа патентных данных.

Основной принцип машинного обучения заключается в представлении данных в виде векторов признаков. Каждый объект в анализируемом наборе данных описывается набором признаков . Эти признаки могут включать текстовые данные патентов, информацию о цитировании, классы Международной патентной классификации (IPC), временные метки и другие релевантные характеристики.

Целевая переменная является той величиной, которую необходимо предсказать. В контексте прогнозирования новых патентных категорий целевая переменная может быть бинарной: , если предсказывается появление новой патентной категории, и , если такой прогноз не подтверждается. Этот бинарный характер задачи делает её задачей классификации.

Методы машинного обучения делятся на два основных класса: регрессия и классификация.

Регрессия в свою очередь может быть линейной и полиномиальной. Линейная регрессия предполагает, что зависимость между целевой переменной и признаками линейна. Модель имеет вид , где – вектор весов, а – смещение. А полиномиальная регрессия расширяет линейную регрессию, включив полиномиальные признаки, что позволяет моделировать нелинейные зависимости.

Классификация имеет множество различных методов реализации. Например, логистическая регрессия применяется для бинарных и многоклассовых задач. Вероятность класса моделируется с использованием логистической функции:

Также достаточно широко используется метод опорных векторов (SVM), который строит гиперплоскость, разделяющую классы, максимизируя зазор между ними.

Еще одним методом машинного обучения с высокой интерпретируемостью является метод деревьев решений, а также ансамблевые методы, которые строят множество деревьев решений и комбинируют их для улучшения точности предсказаний.

В контексте прогнозирования новых технологических направлений, машинное обучение может использоваться для анализа патентов и определения вероятности того, что определённые патентные классы приведут к появлению новых направлений. Примерный процесс выглядит следующим образом:

1. сбор данных о патентах, включая текст описания, дату подачи, цитирование и классы IPC;
2. очистка данных, обработка текстов, векторизация текстовой информации;
3. использование исторических данных для обучения модели классификации, которая предсказывает появление новых патентных классов;
4. оценка точности модели на тестовых данных, корректировка гиперпараметров для улучшения точности.

Методы машинного обучения предоставляют мощные инструменты для прогнозирования в области технологических инноваций. Они позволяют автоматизировать процесс выявления возможных нововведений и могут служить надёжной основой для принятия стратегических решений на основе анализа больших данных. Применение этих методов в анализе патентной информации открывает новые возможности для прогнозирования и управления инновационными процессами.

### 1.4.2 Существующие решения

*Градиентный бустинг деревьев решений*

Градиентный бустинг деревьев решений (Gradient Boosting Decision Trees, GBDT) [28] является ансамблевым методом машинного обучения, который последовательно строит модели, каждая из которых корректирует ошибки предыдущей. Этот метод особенно эффективен для задач регрессии и классификации и широко применяется в анализе больших объемов данных.

GBDT объединяет множество слабых моделей (обычно это деревья решений) в одну сильную модель. В каждом шаге итерации создается новое дерево, которое пытается уменьшить ошибку предыдущих деревьев, добавляя новые прогнозы для улучшения точности. Основной принцип заключается в минимизации функции потерь с использованием градиентного спуска.

GBDT начинается с начальной модели , которая обычно представляет собой среднее значение целевой переменной. На каждом этапе , алгоритм добавляет новое дерево решений , чтобы скорректировать ошибку предыдущей модели. Это выражается следующим образом:

где – это шаг обучения, который контролирует, насколько сильно новое дерево корректирует ошибку.

Цель состоит в минимизации функции потерь , где – реальные значения, а предсказанные значения. В общем виде процесс обучения можно выразить как:

Для регрессии с использованием квадратичной функции потерь (mean squared error, MSE). Это выражается как:

Градиенты вычисляются как:

Далее новые деревья будут основываться на этих градиентах.

GBDT может быть использован для прогнозирования новых технологий на основе анализа патентных данных [29]. Применяя этот метод к данным патентов, можно выявлять тенденции и зависимости, указывающие на возможное появление новых технологических направлений. Например, анализ текстов патентов, их классификаций и временных меток может помочь в прогнозировании будущих технологических инноваций.

Преимуществами GBDT являются:

1. высокая точность предсказаний – GBDT обеспечивает высокую точность благодаря комбинированию множества слабых моделей;
2. способность обрабатывать большие и сложные данные – метод эффективно работает с большими объемами данных и сложными структурами признаков;
3. устойчивость к переобучению при правильной настройке гиперпараметров – возможность настройки параметров, таких как количество деревьев, глубина деревьев и скорость обучения, позволяет контролировать переобучение.

В свою очередь недостатками данного метода являются:

1. высокие вычислительные затраты – обучение GBDT требует значительных вычислительных ресурсов, особенно при большом количестве данных;
2. требуется тщательная настройка гиперпараметров – для достижения оптимальной производительности необходимо тщательно подбирать параметры модели.

GBDT является мощным инструментом для прогнозирования и анализа данных, особенно полезным в контексте патентной информации. Этот метод помогает выявлять скрытые паттерны и тенденции, которые могут указывать на появление новых технологий, что важно для стратегического планирования и инновационной деятельности.

*Случайные леса*

Случайные леса (Random Forest) представляют собой ансамблевый метод машинного обучения, который объединяет множество деревьев решений для повышения точности и стабильности предсказаний. Метод был разработан Лео Брейманом и Адель Катлер в 2001 году [30]. Основная идея случайных лесов заключается в использовании бутстреп-агрегации (bagging) и случайного выбора подмножества признаков для создания некоррелированных деревьев решений. Случайные леса являются достаточно распространенным методом машинного обучения в силу их простой реализации и эффективности решения задач, а также могут применятся в задачах широкого спектра, таких как анализ естественного языка или регрессионные задачи.

Процесс построения случайного леса состоит из нескольких шагов:

1. бутстреп-выборка – из исходного набора данных создаются несколько подвыборок путем случайного выбора с возвращением; это означает, что один и тот же образец может быть выбран несколько раз;
2. построение деревьев – для каждой подвыборки строится дерево решений; при каждом разбиении узла дерева выбирается случайное подмножество признаков; это уменьшает корреляцию между деревьями и помогает снизить переобучение;
3. агрегация результатов – для классификации случайный лес использует метод большинства голосов, где предсказание класса определяется как класс, который получил наибольшее количество голосов от всех деревьев; для регрессии используется среднее значение предсказаний всех деревьев.

Формально, алгоритм случайного леса можно описать следующим образом: создание построение деревьев решений и предсказание.

Построение деревьев решения для :

* cоздать бутстреп-выборку из исходного набора данных ;
* построить дерево решений по бутстреп-выборке , используя случайный выбор подмножества признаков при каждом разбиении узла.

Предсказание для нового наблюдения :

* для классификации: предсказание определяется как класс, который получил наибольшее количество голосов:

где – индикаторная функция;

* для регрессии: предсказание   определяется как среднее значение предсказаний всех деревьев:

Случайные леса могут быть использованы для анализа патентных данных и прогнозирования новых технологических тенденций [31]. Применяя этот метод к данным патентов, можно выявить важные признаки, которые влияют на появление новых технологических направлений, и использовать эту информацию для точного предсказания.

Преимуществами случайных лесов являются:

1. высокая точность и устойчивость к шуму в данных – случайные леса уменьшают вероятность переобучения благодаря использованию множества деревьев и случайного выбора признаков;
2. легкость в интерпретации результатов – метод позволяет оценивать важность каждого признака в предсказаниях модели;
3. эффективность при обработке больших объемов данных – метод хорошо масштабируется и может обрабатывать большие наборы данных с высокой скоростью.

Недостатками случайных лесов являются:

1. медленная работа при большом количестве деревьев – с увеличением числа деревьев растет время вычислений, что может быть проблематично для реального времени;
2. требует больше ресурсов для хранения и обработки данных – для работы метода требуется значительное количество оперативной памяти и вычислительной мощности.

Случайные леса являются мощным и гибким методом машинного обучения, который успешно применяется для прогнозирования и анализа патентных данных. Этот метод помогает выявлять важные признаки и делать точные предсказания, что важно для стратегического планирования и инновационной деятельности.

*Глубокое обучение*

Глубокое обучение (Deep Learning) [32] стало одним из ведущих методов в машинном обучении благодаря своей способности обрабатывать большие объемы данных и выявлять сложные зависимости. В контексте прогнозирования появления новых технологий на основе патентной информации глубокое обучение предоставляет мощные инструменты для анализа текстовых данных, выявления тенденций и предсказания будущих инноваций.

Самые распространенные методы глубокого обучения включают: многослойный перцептрон, свёрточные нейронные сети, рекуррентные нейронные сети, долгая краткосрочная память и архитектура Transformer.

Многослойный перцептрон (MLP) является базовой формой нейронной сети, состоящей из входного слоя, одного или нескольких скрытых слоев и выходного слоя [33]. Каждый нейрон в сети соединен с нейронами следующего слоя, и обучение происходит путем обратного распространения ошибки (backpropagation):

где – предсказание, – входной вектор, – матрица весов, – смещение, – нелинейная активационная функция.

Сеть состоит из нескольких слоев, каждый из которых выполняет свою функцию. Так основными структурными элементами являются:

1. входной слой – каждый нейрон во входном слое представляет одну характеристику входных данных; если у нас есть входных характеристик, то входной слой будет содержать нейронов; входной слой передает данные скрытым слоям без каких-либо преобразований;
2. скрытые слои – позволяют модели учить сложные функции, которые невозможно учить с помощью одного слоя; каждый нейрон в скрытом слое получает входы от всех нейронов предыдущего слоя, применяет весовые коэффициенты и смещения, а затем передает выход через нелинейную активационную функцию;
3. выходной слой – состоит из нейронов, которые производят окончательный вывод модели; количество нейронов в выходном слое зависит от задачи; для задач бинарной классификации используется один нейрон, для многоклассовой классификации – несколько нейронов (по одному на каждый класс).

Для того чтобы сеть могла учить нелинейные зависимости, используются различные функции активации [34] (рисунок 4):

* сигмоидная функция (Sigmoid) – используется для вывода значений в диапазоне от 0 до 1:
* гиперболический тангенс (tanh) – выводит значения в диапазоне от -1 до 1:
* усеченное линейное преобразование (ReLU) – широко используется из-за своей простоты и эффективности в обучении глубоких сетей:

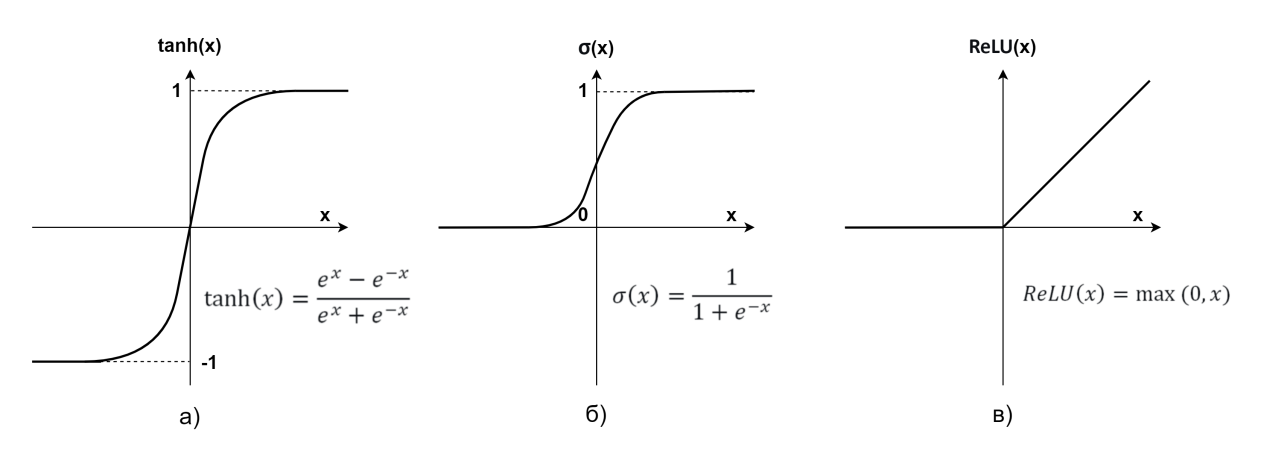


Рисунок 4 – Графики функций активации:

а) – гиперболический тангенс;

б) – сигмоида;

в) – усеченное линейное преобразование

MLP обучается с использованием алгоритма обратного распространения ошибки (backpropagation). Алгоритм включает следующие шаги:

1. прямое распространение (Forward Propagation) – входные данные проходят через сеть, и на каждом слое вычисляется выход;
2. вычисление функции потерь (Loss Calculation) – разница между предсказанными значениями и реальными метками вычисляется с помощью функции потерь, такой как среднеквадратичная ошибка (MSE) или кросс-энтропия;
3. обратное распространение (Backpropagation) – градиенты функции потерь по отношению к параметрам модели (весам и смещениям) вычисляются с использованием метода градиентного спуска;
4. обновление параметров (Parameter Update) – параметры модели обновляются в направлении, противоположном градиенту, чтобы минимизировать функцию потерь:

где – изменение веса, – скорость обучения, – функция потерь, – вес между нейронами и .

MLP может использоваться для различных задач в анализе патентных данных, таких как прогнозирование появления новых технологий, анализ временных рядов публикаций патентов и научных статей для выявления тенденций и предсказания новых технологий. Так в статье [35] описывается методика прогнозирования новых технологий с помощью машинного обучения. В исследовании использовался многослойный перцептрон (MLP) для анализа патентных данных, что позволило выявить новые технологии на ранних стадиях. Методика включает использование различных патентных индикаторов и ассоциативных правил для точного и своевременного обнаружения развивающихся технологий.

Свёрточные нейронные сети (CNN) эффективны для обработки данных с сильной пространственной зависимостью, таких как изображения и текст [36]. Эти сети используют свёрточные слои для автоматического извлечения признаков из входных данных, что делает их мощным инструментом для анализа текстовых данных патентов и выявления новых технологических тенденций.

Сверточная сеть состоит из сверточных слоев, субдеискретизации, полносвязных слоев.

Свёрточные слои являются основными структурными элементами CNN. Они применяют фильтры (также называемые ядрами) к входным данным для извлечения признаков. В контексте текста фильтры применяются к последовательностям слов или символов, выявляя ключевые фразы или комбинации слов:

где – значение на выходе свёрточного слоя, – входные данные, – фильтр, и – смещение.

Субдискретизация (Pooling) используется для уменьшения размерности данных, сохраняя при этом важную информацию. Это помогает уменьшить вычислительные затраты и предотвратить переобучение. Среди используемых методов субдескритезации выделяют:

* max pooling – выбирает максимальное значение из окна признаков;
* mean pooling – вычисляет среднее значение признаков в окне;
* k-max pooling – выбирает максимальных значений, сохраняя порядок.

Полносвязные слои используются в конце сети для интеграции извлечённых признаков и выполнения финальной классификации или регрессии. Каждый нейрон в полносвязном слое соединён со всеми нейронами предыдущего слоя.

CNN могут быть использованы для анализа текстовых данных патентов для выделения ключевых слов и фраз, которые указывают на новые технологические тенденции. Они могут эффективно классифицировать патенты по категориям и прогнозировать вероятные направления развития технологий. Так в статье [37] рассматривается методика прогнозирования появления новых технологий с использованием сверточных нейронных сетей (CNN) и многослойных перцептронов (MLP). Исследование использует 18 входных и 3 выходных индикатора из базы данных Патентного ведомства США. Эти методы машинного обучения позволяют выявлять сложные нелинейные взаимосвязи и анализировать тенденции развития технологий на ранних стадиях, что помогает в стратегическом планировании и прогнозировании технологических трендов.

Рекуррентные нейронные сети (RNN) используются для работы с последовательными данными, такими как временные ряды и текст. Они имеют циклические связи, которые позволяют учитывать предыдущие состояния для предсказания текущих [38].

Cхематическое представление RNN приведено на рисунке 5. Основное уравнение RNN описывает, как вычисляется скрытое состояние в момент времени :

где – скрытое состояние в момент времени , – входной вектор, и – матрицы весов, – смещение, – нелинейная функция активации (например, ReLU, Sigmoid или Tanh).

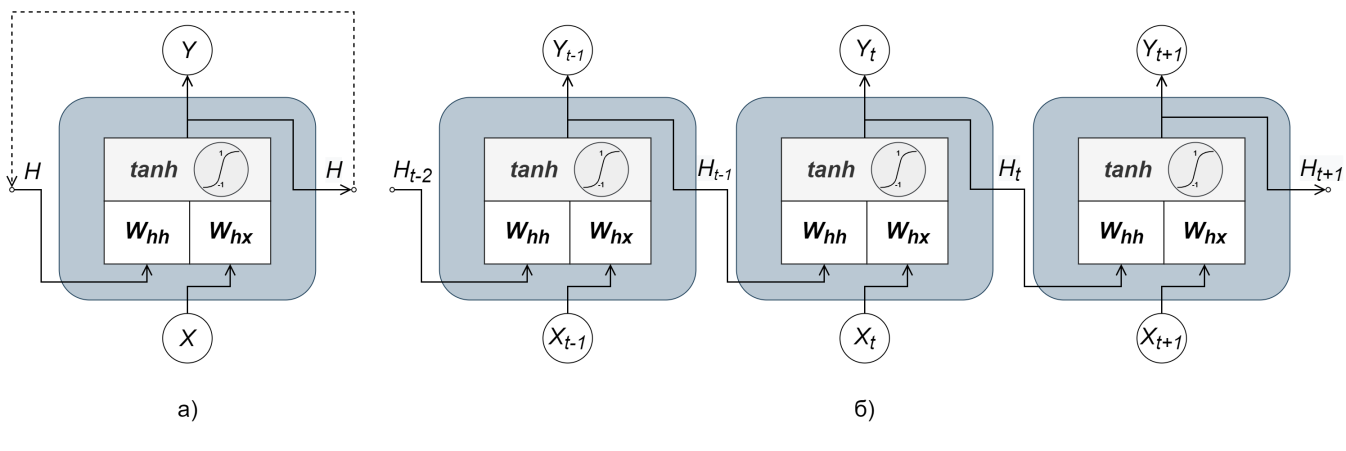


Рисунок 5 – Концептуальные схемы:

а) – ячейка RNN, б) – цепочка ячеек RNN

Обучение RNN осуществляется с использованием алгоритма обратного распространения ошибки через время (BPTT, Backpropagation Through Time). Этот алгоритм включает в себя развёртывание сети во времени и вычисление градиентов функции потерь по отношению к параметрам модели.

Типы RNN (рисунок 6):

1. многие ко многим (Many-to-Many) – например, машинный перевод, где входная последовательность (предложение на одном языке) преобразуется в выходную последовательность (предложение на другом языке);
2. многие к одному (Many-to-One) – например, анализ настроений, где последовательность слов преобразуется в один выход (положительное или отрицательное настроение);
3. один ко многим (One-to-Many) – например, генерация текста, где один входной токен запускает генерацию последовательности выходных токенов.

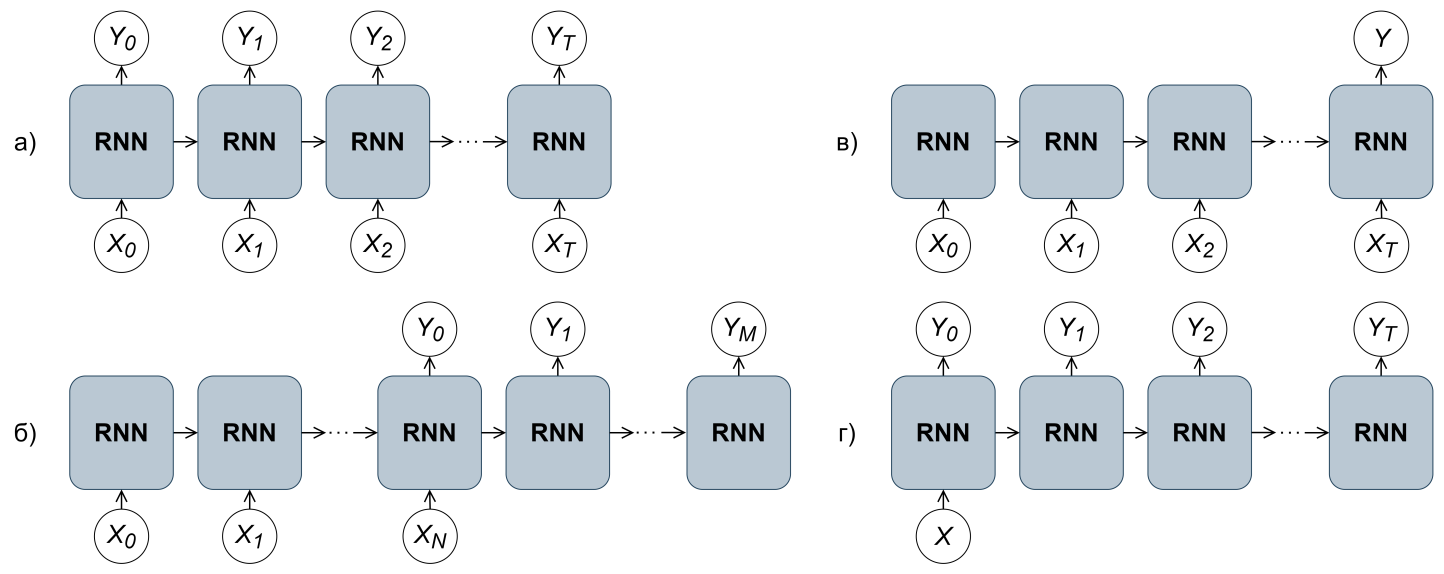


Рисунок 6 – Типы ячеек RNN:

а, б) – Many-to-Many;

в) – Many-to-One;

г) – One-to-Many

RNN могут использоваться для анализа временных рядов публикаций патентов и научных статей, выявления тенденций и предсказания новых технологий. Важное применение RNN в контексте патентов – это прогнозирование появления новых технологических направлений на основе анализа временных последовательностей данных.

Так, в статье [39] также включается использование рекуррентных нейронных сетей (RNN) и многослойных перцептронов (MLP) для анализа патентных данных, что помогает точно и своевременно идентифицировать новые технологии на ранних стадиях их развития.

Долгая краткосрочная память (LSTM) – это тип рекуррентной нейронной сети (RNN), специально разработанный для решения проблемы затухающих градиентов, характерной для традиционных RNN [40]. Эта проблема возникает, когда градиенты, используемые для обучения сети, становятся очень маленькими, что приводит к остановке обучения. LSTM была введена в 1997 году Шеппом Хохрайтером и Юргеном Шмидхубером и с тех пор была улучшена и популяризована.

LSTM способна запоминать долгосрочные зависимости за счет использования специальных ячеек памяти и трех типов ворот: входные, выходные и забывающие (рисунок 7).

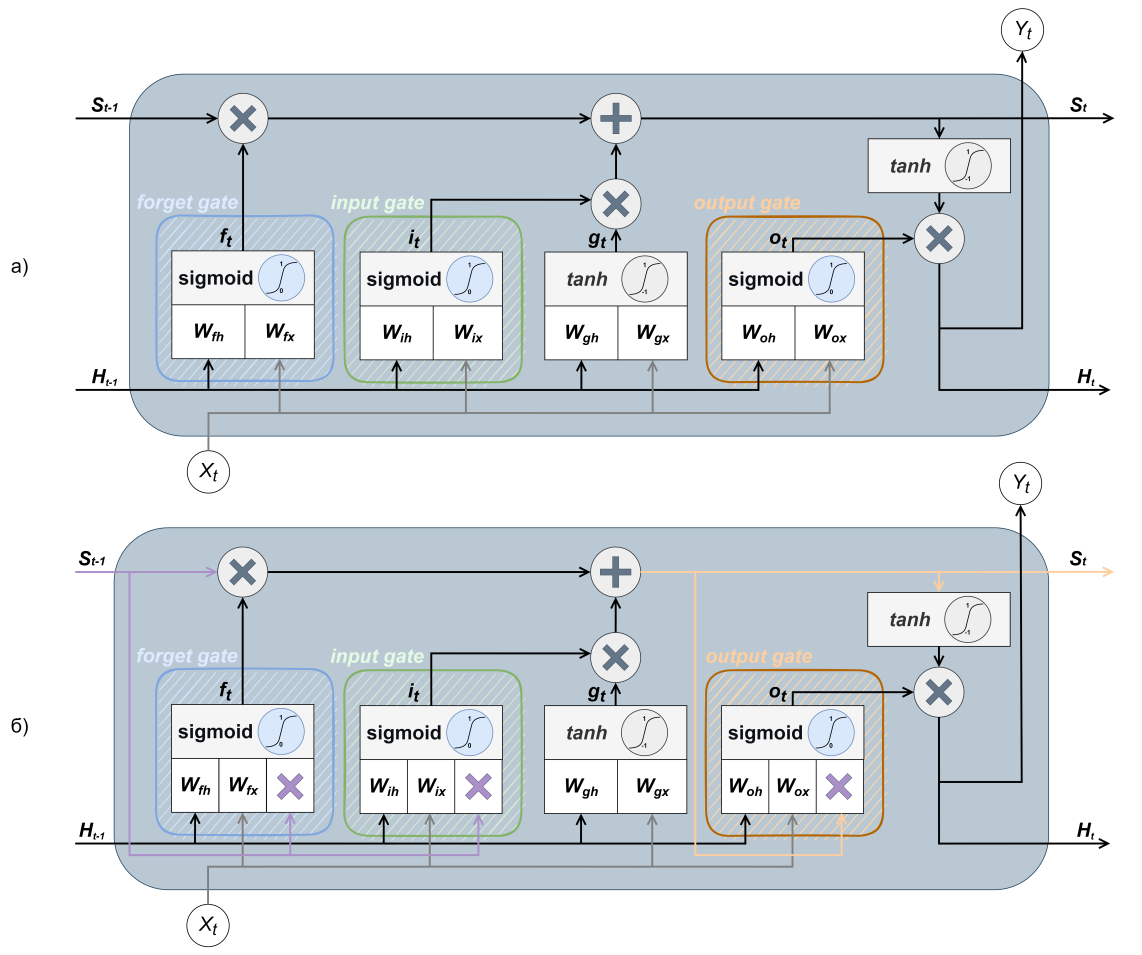


Рисунок 7 – Схема ячейки LSTM

LSTM состоит из следующих структурных элементов:

1. ячейка памяти – является основным элементом LSTM, который позволяет сохранять информацию на протяжении длительного времени; она действует как конвейер, через который может проходить информация с минимальными изменениями; ячейка памяти обновляется по формуле:
2. входные ворота (Input Gate) – определяют, какая информация из текущего входа и предыдущего скрытого состояния будет добавлена в ячейку памяти:
3. забывающие ворота (Forget Gate) – решают, какая информация из ячейки памяти будет удалена; они применяют сигмоидную функцию для решения, что забыть и что сохранить :
4. выходные ворота (Output Gate) – определяют, какая информация из ячейки памяти будет использована для вычисления выходного состояния:

где , , – значения ворот, – состояние памяти, – скрытое состояние, и – сигмоида.

LSTM может использоваться для анализа временных рядов публикаций патентов и научных статей. Это позволяет выявлять тенденции и прогнозировать появление новых технологий, анализируя последовательные данные. Например, можно использовать LSTM для предсказания количества патентов, поданных в определенной области технологии за следующий год, на основе исторических данных.

Так, в статье [41] посвящена анализу жизненного цикла технологий с использованием метода долгосрочной краткосрочной памяти (LSTM). В исследовании предлагается новая методика прогнозирования появления новых технологий путем анализа патентных данных. Применение LSTM позволяет более точно и эффективно выявлять тенденции развития технологий на разных этапах их жизненного цикла, начиная с появления и заканчивая стадиями роста, зрелости и насыщения.

Архитектура трансформер была предложена в 2017 году в статье «Attention Is All You Need» [42] и быстро стала основой для многих современных моделей обработки естественного языка (NLP). В отличие от рекуррентных и свёрточных нейронных сетей, трансформер полностью устраняет рекуррентные связи и использует механизмы внимания для параллельной обработки данных, что значительно ускоряет обучение и улучшает производительность.

Архитектура состоит из энкодеров и декодеров, вводных и позиционных кодировок, механизма самовнимания, многоголовного самовнимания, полносвязных слоев и нормализации.

Энкодеры и декодеры (рисунок 8). В оригинальной статье используется по шесть энкодеров и декодеров, но их количество можно варьировать. Энкодеры преобразуют входные последовательности в контекстуализированные представления, а декодеры используют эти представления для генерации выходных последовательностей.

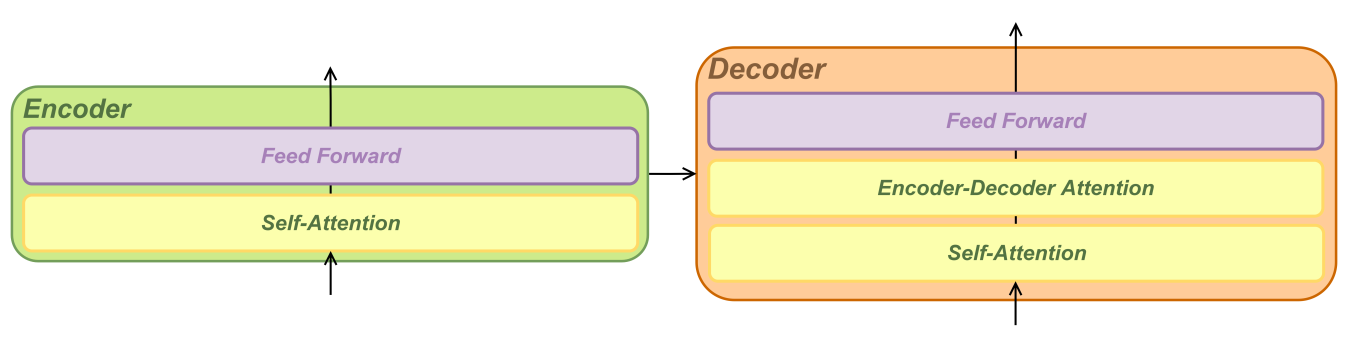


Рисунок 8 – Энкодер и декодер в архитектуре трансформера

Вводные и позиционные кодировки. Каждое слово или токен преобразуется в векторное представление с помощью слоев embedding. Поскольку трансформер не имеет рекуррентных связей, позиционные кодировки добавляются к векторным представлениям, чтобы учитывать порядок слов в предложении. Это достигается с помощью синусоидальных и косинусоидальных функций.

Механизм самовнимания (Self-Attention). Самовнимание позволяет модели учитывать все слова в последовательности при обработке каждого слова. Механизм использует матрицы запросов (), ключей () и значений ():

где – запросы (queries), – ключи (keys), – значения (values), и – размерность ключей.

Многоголовое внимание (Multi-Head Attention). Многоголовое внимание улучшает способность модели фокусироваться на различных частях входных данных. Это достигается путем использования нескольких наборов матриц , и , которые обучаются независимо и затем объединяются.

Полносвязные слои и нормализация. Каждый энкодер и декодер включает полносвязные слои и слои нормализации для стабилизации обучения и улучшения производительности.

Архитектура трансформеров представляет собой значительный шаг вперед в области машинного обучения, особенно в задачах обработки естественного языка. Благодаря своей эффективности и гибкости, она стала основой для многих современных моделей, таких как GPT и BERT, и продолжает находить новые применения, включая анализ патентных данных и прогнозирование новых технологических направлений.

Например, в статье [43] основное внимание уделяется выявлению тенденций и подбору партнеров по сотрудничеству, учитывая взаимозависимость знаний и сотрудничества. В исследовании применяются модели трансформеров для анализа сетей сотрудничества и знаний, что позволяет организациям стратегически выбирать партнеров для инновационных проектов и прогнозировать будущие технологические тренды. Использование трансформеров помогает более точно анализировать большие объемы данных и улучшить процесс принятия решений, что способствует повышению эффективности инновационной деятельности.

Глубокое обучение может быть использовано для анализа патентных данных и предсказания будущих технологических направлений. Например, LSTM-модели могут анализировать временные ряды публикаций патентов и научных статей для выявления тенденций и предсказания новых технологий.

Свёрточные нейронные сети и Transformer могут использоваться для кластеризации и классификации патентов по тематикам. Это позволяет идентифицировать основные направления. Глубокое обучение эффективно работает с текстовыми данными, что позволяет анализировать содержание патентов, выделять ключевые слова и фразы, и выявлять новые технологические тенденции.

Преимуществами методов глубокого обучения являются:

1. способность обрабатывать и анализировать неструктурированные данные – глубокое обучение позволяет анализировать большие объемы текстовых данных, таких как патенты;
2. высокая точность предсказаний – модели глубокого обучения достигают высокой точности при наличии большого количества данных;
3. способность выявлять сложные и нелинейные зависимости – эти модели могут обнаруживать сложные паттерны и взаимосвязи в данных.

Недостатками являются:

1. требуют большого количества данных для обучения – для эффективного обучения моделей глубокого обучения необходимо много данных;
2. высокие вычислительные затраты – обучение и использование моделей глубокого обучения требуют значительных вычислительных ресурсов;
3. сложности в настройке и интерпретации – модели глубокого обучения часто сложны для настройки и интерпретации результатов.

Глубокое обучение представляет собой хороший инструмент для прогнозирования новых технологий на основе патентных данных. Применение моделей глубокого обучения позволяет анализировать большие объемы данных, выявлять скрытые паттерны и делать точные предсказания, что важно для стратегического планирования и инновационной деятельности.

*Ассоциационные правила*

Метод ассоциационных правил (Association Rule Mining, ARM) используется для выявления частых шаблонов, ассоциаций и корреляций между элементами в больших наборах данных [44]. Этот метод особенно полезен для анализа транзакционных данных, но также находит применение в анализе патентных данных для прогнозирования новых технологических направлений. ARM позволяет выявлять скрытые связи между различными патентами и предсказывать появление новых технологий на основе частоты совместного упоминания определенных технологий и их комбинаций.

Ассоциационные правила описываются в виде импликаций , где и – подмножества элементов. Правило читается как «если присутствует в транзакции, то также вероятно присутствует в этой транзакции».

Основные метрики для оценки ассоциационных правил:

1. Support (Поддержка) – показывает, как часто элемент или набор элементов встречается в базе данных;
2. Confidence (Достоверность) – достоверность измеряет, насколько часто правило оказывается истинным;
3. Lift (Поддержка) – оценивает, насколько часто и появляются вместе по сравнению с тем, если бы они были независимы.

Для нахождения ассоциационных правил применяются следующие алгоритмы:

1. алгоритм Apriori – использует итеративный подход для поиска частых наборов элементов; он использует свойство, что если набор элементов является частым, то все его подмножества также часты; этот алгоритм эффективно сокращает количество наборов, которые необходимо рассмотреть, за счет удаления нечастых наборов;
2. алгоритм FP-Growth (Frequent Pattern Growth) улучшает алгоритм Apriori, устраняя необходимость в генерации кандидатов; вместо этого он строит дерево частых шаблонов (FP-tree) и использует его для обнаружения частых наборов элементов.

Преимуществами данного метода являются:

* способность выявлять скрытые паттерны в больших наборах данных;
* простота в реализации и интерпретации;
* полезность для выявления связей и зависимостей между объектами.

Недостатки:

* генерация большого количества правил, которые сложно интерпретировать;
* чувствительность к минимальным уровням поддержки и достоверности, что может повлиять на качество результатов.

Ассоциационные правила могут использоваться для анализа патентов и выявления новых технологических направлений. Например, частое совместное упоминание определенных технологических терминов в патентах может указывать на возникновение новых технологических тенденций.

Так, в статье [45] описывается методика анализа патентов с целью прогнозирования появления новых технологий. В исследовании используются методы поиска ассоциативных правил и моделирования по Боксу-Дженкинсу для выявления закономерностей в данных патентов. Это позволяет предсказывать технологические тренды и выявлять новые перспективные технологии. Основное внимание уделяется анализу данных Европейского патентного ведомства с помощью кластерного анализа, текстового майнинга и социальных сетевых методов, что позволяет формировать информативные технологические карты и улучшать стратегическое планирование инноваций.

*Байесовские сети*

Байесовские сети (Bayesian Networks) – это графические модели, которые используют вероятностные зависимости между переменными для моделирования сложных систем и прогнозирования. В контексте анализа патентных данных и прогнозирования новых технологий байесовские сети могут выявлять скрытые зависимости между различными патентными классами и предсказывать появление новых технологических направлений.

Байесовские сети представляют собой направленные ациклические графы (DAG), где узлы соответствуют переменным, а ребра – условным зависимостям между ними. Каждой переменной присваивается условное распределение вероятностей , где – это множество родительских узлов для .

Совместное распределение всех переменных в байесовской сети можно выразить как произведение всех условных распределений:

Одним из основополагающих понятий в байесовских сетях является условная независимость. Переменные и условно независимы, если:

где множество переменных, обеспечивающих независимость и .

Преимуществами такого подхода являются:

* способность моделировать и интерпретировать сложные зависимости – байесовские сети позволяют визуализировать и анализировать сложные зависимости между переменными;
* поддержка как количественных, так и качественных данных – модель может работать с различными типами данных, что делает её универсальной;
* высокая точность предсказаний при правильной настройке модели – модели могут быть очень точными, если правильно настроены и обучены.

Недостатки:

* требует значительных вычислительных ресурсов для обучения и построения сети – процесс обучения может быть вычислительно затратным, особенно для больших сетей;
* сложность в интерпретации и настройке модели: модели могут быть сложными для настройки и интерпретации, требуя глубоких знаний в области вероятностного моделирования.

Байесовские сети могут использоваться для моделирования зависимостей между различными патентными классами и прогнозирования появления новых технологий на основе этих зависимостей. Например, можно построить модель, где узлы представляют патентные классы, а ребра – вероятностные зависимости между ними. Используя исторические данные о патентах, можно обучить модель и использовать её для предсказания новых технологических направлений.

Так, в работе [46] описывается методика создания технологических дорожных карт, которые не только адаптируются к меняющимся обстоятельствам, но и прогнозируют появление новых технологий. Авторы используют байесовские сети и анализ тематического моделирования для выявления и оценки рисков, что позволяет динамически корректировать планы и прогнозировать новые технологические тенденции. Этот подход был успешно применен в области искусственного интеллекта, демонстрируя его эффективность и устойчивость в управлении технологиями.

### 1.4.3 Сравнительный анализ

Сравнение и краткий анализ рассмотренных методов приведен в таблице 3.

Таблица 3 – Сравнение методов на основе машинного обучения.

| **Метод** | **Преимущества** | **Недостатки** | **Применение** |
| --- | --- | --- | --- |
| GBDT | Высокая точность, устойчивость к переобучению, способность обрабатывать большие данные | Высокие вычислительные затраты, требуется настройка гиперпараметров | Прогнозирование технологических направлений, анализ текстов патентов |
| Random Forest | Высокая точность, устойчивость к шуму, интерпретируемость | Медленная работа при большом числе деревьев, высокие требования к ресурсам | Выявление важных признаков, прогнозирование технологических тенденций |
| MLP | Способность обучаться нелинейным зависимостям, высокая точность | Требует больших данных для обучения, склонность к переобучению | Прогнозирование появления новых технологий, анализ временных рядов публикаций |
| CNN | Эффективны для обработки данных с пространственной зависимостью, высокая точность | Требуют больших вычислительных ресурсов, сложность настройки | Анализ текстовых данных патентов, выявление ключевых слов и фраз |
| RNN | Эффективны для последовательных данных, учитывают временные зависимости | Проблема затухающих градиентов, сложность обучения | Прогнозирование на основе временных рядов публикаций патентов |
| LSTM | Решение проблемы затухающих градиентов, запоминание долгосрочных зависимостей | Высокие вычислительные затраты, сложность настройки | Анализ временных рядов публикаций, прогнозирование новых технологий |

Продолжение таблицы 3

|  |  |  |  |
| --- | --- | --- | --- |
| **Метод** | **Преимущества** | **Недостатки** | **Применение** |
| Transformer | Высокая производительность, параллельная обработка данных, эффективность в задачах NLP | Требуют больших данных и вычислительных ресурсов, сложность архитектуры | Классификация и кластеризация патентов, анализ сетей сотрудничества и знаний |
| Ассоциационные правила | Выявление скрытых паттернов, простота реализации и интерпретации | Генерация большого числа правил, чувствительность к уровням поддержки и достоверности | Выявление закономерностей в патентах, прогнозирование технологических направлений на основе частоты упоминаний |
| Байесовские сети | Моделирование сложных зависимостей, поддержка количественных и качественных данных | Высокие вычислительные затраты, сложность интерпретации и настройки | Моделирование зависимостей между патентами, прогнозирование новых технологических направлений |

Методы глубокого обучения предоставляют качественные инструменты для анализа и прогнозирования на основе больших данных. Каждый из рассмотренных методов имеет свои преимущества и недостатки, что делает их подходящими для различных типов задач:

* GBDT и Random Forest обеспечивают высокую точность и устойчивость к шуму, но требуют значительных вычислительных ресурсов и тщательной настройки;
* MLP и CNN эффективны для анализа данных с нелинейными зависимостями и признаками, но также требуют больших объемов данных и вычислительных затрат;
* RNN и LSTM особенно полезны для последовательных данных и временных рядов, хотя имеют проблемы с затухающими градиентами.

Архитектура трансформеров представляет собой современный подход, обеспечивающий высокую эффективность в задачах обработки естественного языка, но также требует значительных ресурсов.

## 1.5 Методы векторизации текста патентных документов

Для большинства современных методов и подходов к прогнозированию, таких как рекуррентные нейронные сети (RNN), свёрточные нейронные сети (CNN) и архитектуры типа Transformer, текстовые данные являются основным источником информации [47]. В контексте анализа патентных данных тексты патентов содержат ценную информацию о технологических инновациях, тенденциях и направлениях развития. Однако текстовые данные, особенно в форме патентов, не подходят для прямого использования в моделях машинного обучения из-за их неструктурированной природы.

Текстовые данные состоят из последовательностей слов, предложений и абзацев, которые сложно интерпретировать напрямую с помощью алгоритмов машинного обучения. Для того чтобы извлечь полезную информацию из текстов и использовать её в предсказательных моделях, необходимо преобразовать тексты в формализованные представления – векторы. Векторизация текстовых данных позволяет представить слова и фразы в виде числовых векторов, которые могут быть легко обработаны различными алгоритмами. Эти методы подробно рассмотрены в статье [48], далее изложено краткое описание концепции методов.

### 1.5.1 TF-IDF

TF-IDF (Term Frequency-Inverse Document Frequency) – это популярный метод векторизации текстовых данных, широко используемый в информационном поиске и текстовом анализе. Этот метод оценивает важность слова в документе относительно всего корпуса документов. Он основан на двух основных компонентах: частоте термина (TF) и обратной частоте документа (IDF).

Частота термина – это количество раз, которое слово встречается в документе, деленное на общее количество слов в этом документе. Это позволяет нормализовать количество вхождений слова, чтобы длинные документы не имели преимущества перед короткими:

где – количество вхождений термина в документ , – общее количество слов в документе .

Обратная частота документа измеряет важность термина в корпусе документов. Она рассчитывается как логарифм отношения общего числа документов к количеству документов, содержащих данный термин:

где – общее количество документов, – количество документов, содержащих термин .

Итоговый вес термина рассчитывается как произведение TF и IDF, что позволяет учитывать как локальную важность слова в документе, так и его глобальную редкость в корпусе:

Перед применением TF-IDF важно провести предобработку текста, которая включает следующие шаги:

1. удаление стоп-слов – стоп-слова (например, «и», «но», «или») не несут значимой информации и должны быть удалены. Это можно сделать с помощью списков стоп-слов или определенных порогов частоты слов;
2. токенизация – разбиение текста на отдельные слова или токены;
3. нормализация – приведение слов к нижнему регистру для устранения различий между словами, написанными с большой или маленькой буквы;
4. лемматизация и стемминг – приведение слов к их начальной форме (лемма) или основе (стем).

TF-IDF широко используется в задачах классификации текста, извлечения ключевых слов и информации, а также в информационно-поисковых системах. Метод позволяет эффективно учитывать как частоту термина в конкретном документе, так и его значимость в целом корпусе, что делает его хорошим инструментом для анализа текстов.

### 1.5.2 Bag of Words

Bag of Words (BoW) – это один из самых простых и широко используемых методов векторизации текстовых данных. Он представляет текст как «мешок слов», игнорируя порядок слов и учитывая лишь их наличие и частоту. Несмотря на свою простоту, этот метод часто используется в задачах обработки естественного языка (NLP) и информационного поиска.

Метод Bag of Words преобразует текстовые данные в числовые векторы, что позволяет применять различные алгоритмы машинного обучения для анализа текстов. В BoW модель каждая текстовая единица (документ) представлена как вектор длины, равной размеру словаря, где каждый элемент вектора указывает количество вхождений соответствующего слова в документе:

где – частота слова в документе .

Аналогично TF-IDF для данного метода требуется предварительная обработка текста.

### 1.5.3 Word Embeddings

Word embeddings – это метод представления текстовых данных в виде плотных векторов чисел. В отличие от методов, таких как TF-IDF или Bag of Words, которые создают разреженные и высокоразмерные векторы, word embeddings позволяют получить низкоразмерные плотные векторы, которые улавливают семантические связи между словами. Эти векторы обучаются таким образом, что слова с похожим значением имеют близкие векторные представления в многомерном пространстве.

Основная идея word embeddings основана на распределительной гипотезе, которая гласит, что слова, имеющие похожие контексты, также имеют похожие значения. Для создания таких представлений используются нейронные сети, которые обучаются на больших объемах текстовых данных.

Примерами реализации концепции word embeddings являются алгоритмы:

1. Word2Vec – разработан в Google, включает две модели: Continuous Bag of Words (CBOW) и Skip-Gram; CBOW предсказывает целевое слово на основе контекстных слов, тогда как Skip-Gram делает наоборот – предсказывает контекстные слова на основе целевого слова:

где – корпус текстов, – целевое слово, – контекстное слово, – вероятность контекстного слова при заданном целевом слове , вычисляемая через softmax функцию;

1. GloVe (Global Vectors for Word Representation) – разработан в Стэнфорде, обучается на глобальной статистике соположения слов, создавая векторы, которые учитывают частоту совместного появления слов в корпусе текстов.

### 1.5.4 Doc2Vec

Doc2Vec – это расширение метода Word2Vec, разработанное для создания векторных представлений целых документов, а не отдельных слов. Этот метод позволяет эффективно представлять текстовые данные в виде фиксированных векторов, что полезно для задач классификации, кластеризации и поиска по тексту.

Doc2Vec использует нейронные сети для обучения векторных представлений документов. Основная идея заключается в том, чтобы представлять каждый документ как уникальный вектор, обучаемый вместе с векторами слов, что позволяет учитывать как локальные, так и глобальные контекстуальные зависимости.

Существует две основные модели Doc2Vec:

1. Distributed Memory Model of Paragraph Vectors (PV-DM) – эта модель аналогична модели Continuous Bag of Words (CBOW) в Word2Vec; в PV-DM модель предсказывает слово, основываясь на контексте слов и векторе документа:

где – выходной вектор слова , ( – контекстные слова, – вектор документа, – все веса в словаре;

1. Distributed Bag of Words version of Paragraph Vector (PV-DBOW) – эта модель аналогична модели Skip-Gram в Word2Vec; в PV-DBOW модель предсказывает контекстные слова, используя вектор документа:

где – слова в контексте документа .

### 1.5.5 Sentence Transformers

Sentence Transformers, или SBERT, – это современный метод преобразования предложений в плотные векторные представления, разработанный на основе архитектуры BERT (Bidirectional Encoder Representations from Transformers). Этот метод был представлен в статье «Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks» и получил широкое распространение благодаря своей эффективности в задачах вычисления семантического сходства, классификации текстов и кластеризации.

Основная идея SBERT заключается в использовании модели BERT для создания векторных представлений предложений. В отличие от оригинальной модели BERT, которая обрабатывает каждое предложение отдельно, SBERT обучается на парах предложений, используя архитектуру типа «Сиамская сеть». Это позволяет модели учитывать семантические связи между предложениями и создавать более точные векторные представления.

Процесс векторизации:

1. токенизация – каждое предложение разбивается на токены, которые затем подаются на вход модели BERT;
2. вычисление эмбеддингов – для каждого токена вычисляется его эмбеддинг с помощью многослойной трансформерной сети;
3. среднее пуллирование – для получения фиксированного векторного представления предложения используется среднее пуллирование эмбеддингов всех токенов.

### 1.5.6 Сравнение методов векторизации текста патентов

Сравнение рассмотренных методов приведено в таблице 4.

Таблица 4 – Сравнительный анализ методов векторизации

| **Метод** | **Преимущества** | **Недостатки** | **Применение** |
| --- | --- | --- | --- |
| TF-IDF | Простота реализации, эффективность для начального анализа | Игнорирует порядок слов, создает разреженные векторы | Полезен для базового анализа и извлечения ключевых слов |
| Bag of Words (BoW) | Легкость в реализации и интерпретации | Потеря контекста, разреженные векторы, высокая размерность | Подходит для простых задач классификации и начального анализа |
| Word Embeddings | Клавливает семантические связи между словами, плотные векторы | Требует большого объема данных для обучения, игнорирует структуру документа | Хорошо подходит для более сложных задач анализа текста |
| Doc2Vec | Учитывает контекст документов, создает плотные векторы | Сложнее в реализации, требует значительных вычислительных ресурсов | Эффективен для задач классификации и кластеризации документов |
| Sentence Transformers | Высокая точность и производительность, улавливает сложные зависимости | Требует мощных вычислительных ресурсов, сложность в обучении | Идеален для глубокого анализа текста и выявления новых технологий |

Каждый метод векторизации имеет свои преимущества и недостатки, и выбор метода зависит от конкретной задачи анализа текстов патентов. Для базового анализа и извлечения ключевых слов могут быть полезны TF-IDF и BoW. Для более сложного анализа и прогнозирования новых технологий лучше подойдут Word Embeddings, Doc2Vec и Sentence Transformers.

## Выводы

В исследовательской части выполнен анализ различных подходов к прогнозированию появления новых технологий на основе патентной информации. Рассмотренные методы включают кластеризацию патентов, использование ресурсных сетей и машинное обучение. Каждый из этих подходов имеет свои особенности и преимущества, которые помогают в выявлении и предсказании новых технологических направлений.

Первым рассмотренным методом является кластеризация патентов, которая позволяет группировать патенты по схожим характеристикам. Применение таких алгоритмов, как K-means, иерархическая кластеризация и аффинная пропагация, позволяет эффективно выявлять основные технологические направления. Анализ структуры кластеров во времени помогает выявить рост или сокращение интереса к определенным технологическим областям, а также появление новых кластеров, что может свидетельствовать о зарождении новых технологических направлений. Важным элементом этого подхода является преобразование текстовой информации патентов в числовые векторы с помощью методов TF-IDF, Word2Vec и контекстуальных моделей, таких как BERT и GPT-3.

Подход на основе ресурсных сетей представляет собой методологию, направленную на моделирование распространения ресурсов по графу, где вершины представляют патентные классы или технологические области, а ребра – связи между ними. Использование марковских цепей и скрытых марковских моделей (HMM) позволяет прогнозировать появление новых объектов патентного классификатора на основе анализа текущего состояния графа и достижения пороговых значений ресурса в вершинах. Этот метод эффективен для моделирования динамических систем и анализа сложных зависимостей между патентами и технологическими направлениями.

Методы машинного обучения играют важную роль в прогнозировании новых технологических направлений. В частности, алгоритмы регрессии и классификации, такие как логистическая регрессия, метод опорных векторов (SVM), деревья решений и ансамблевые методы (GBDT и случайные леса), позволяют анализировать большие объемы данных и делать точные предсказания. Глубокое обучение, включающее многослойные перцептроны (MLP), свёрточные нейронные сети (CNN) и рекуррентные нейронные сети (RNN), обеспечивает достойные инструменты для обработки текстовых данных и выявления сложных взаимосвязей, что особенно важно для анализа патентных данных и прогнозирования технологических трендов.

Каждый из рассмотренных методов имеет свои уникальные преимущества и недостатки. Например, методы кластеризации просты в реализации и позволяют легко визуализировать результаты, но могут быть чувствительны к выбросам и требуют задания числа кластеров. Ресурсные сети и марковские модели эффективны для анализа временных зависимостей и прогнозирования на основе сложных вероятностных структур, однако требуют значительных вычислительных ресурсов и точной настройки параметров. Машинное обучение и глубокие нейронные сети обеспечивают высокую точность предсказаний и способность обрабатывать большие данные, но требуют большого объема данных для обучения и значительных вычислительных мощностей.

Сравнительный анализ методов показывает, что комбинирование различных подходов может привести к наилучшим результатам в прогнозировании появления новых технологий. Использование кластеризации для предварительного анализа данных, применение ресурсных сетей для моделирования динамических процессов и использование машинного обучения для точного предсказания новых направлений позволяет создать комплексную систему, способную эффективно прогнозировать технологические тренды.

Эффективная визуализация результатов играет важную роль в интерпретации сложных взаимосвязей между патентами и технологическими областями. Для этого используются инструменты, такие как дендрограммы, тепловые карты и сетевые графы, которые помогают исследователям лучше понимать структуру данных и выявлять ключевые тенденции.

В заключение, методы анализа патентной информации и прогнозирования новых технологий на основе кластеризации, ресурсных сетей и машинного обучения предоставляют хорошие инструменты для стратегического планирования и управления инновационной деятельностью. Они позволяют не только анализировать текущее состояние научно-технического развития, но и предсказывать возможные направления будущего развития, что имеет важное значение для принятия обоснованных решений в области научных исследований и инноваций.

# 2 Проектирование и реализация программного продукта

## 2.1 Разработка архитектуры программной системы

### 2.1.1 Анализ вариантов построения систем

При разработке программной системы прогнозирования появления новых технологических областей на основе патентной информации крайне важно выбрать подходящую архитектуру. Архитектура системы определяет ее структуру, масштабируемость, гибкость и способность к модернизации. В этом разделе рассматривается три основных архитектурных подхода, которые могут быть использованы для создания такой системы: монолитная архитектура, сервис-ориентированная архитектура и микросервисная архитектура [49]. Анализ этих вариантов позволит определить наиболее подходящий подход для реализации системы, обеспечивая эффективное выполнение всех ее функциональных требований.

*Монолитная архитектура*

Монолитная архитектура представляет собой традиционный подход к разработке программного обеспечения, в котором все компоненты системы объединены в единое целое. В контексте программной системы прогнозирования появления новых технологических областей на основе патентной информации, использование монолитной архитектуры имеет свои особенности, преимущества и недостатки [50].

На рисунке 9 изображен пример монолитной архитектуры.

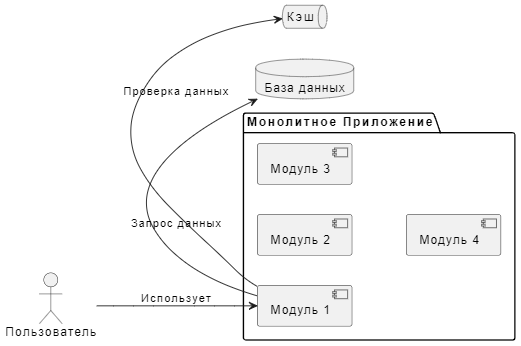


Рисунок 9 – Пример монолитной архитектуры

Монолитная архитектура характеризуется:

1. единым приложением – все функциональные модули, включая парсинг патентных документов, создание и обучение моделей, визуализацию данных и генерацию отчетов, находятся в одном приложении; это означает, что все компоненты системы работают в одном процессе и исполняются на одном сервере;
2. централизованном управление – в монолитной архитектуре все части системы управляются централизованно, что упрощает разработку и развертывание; одно изменение в коде может затронуть несколько модулей одновременно, что требует тщательного тестирования перед выпуском новых версий;
3. общей базой данных – монолитные приложения обычно используют одну общую базу данных для всех компонентов; это упрощает управление данными, но может стать узким местом при масштабировании системы.

Преимущества монолитной архитектуры:

1. простота разработки – поскольку все компоненты системы объединены в одном приложении, разработчикам легче управлять кодовой базой; все зависимости и взаимодействия между модулями видны и контролируемы;
2. упрощенное развертывание – развертывание монолитного приложения является достаточно простым процессом, так как нужно работать только с одним исполняемым файлом или пакетом; это снижает сложность инфраструктуры и упрощает процесс обновлений;
3. транзакционная целостность – в монолитной архитектуре легче обеспечить целостность данных, так как все операции происходят в рамках одного приложения и одной базы данных; это особенно важно для систем, работающих с критически важной информацией, такой как патентные данные.

Недостатки монолитной архитектуры:

1. ограниченная масштабируемость – масштабирование монолитного приложения может быть сложным, так как для увеличения производительности необходимо масштабировать всю систему целиком; это может привести к избыточному потреблению ресурсов и снижению эффективности;
2. сложность управления – по мере роста приложения и увеличения числа функциональных модулей управление монолитным кодом становится все более сложным; внесение изменений в одном модуле может непредсказуемо повлиять на другие модули, что увеличивает риск возникновения ошибок;
3. трудности в поддержке и модернизации – в случае монолитной архитектуры модернизация системы и добавление новых функций требуют тщательного планирования и тестирования; из-за высокой связанности компонентов любое изменение может вызвать неожиданные проблемы в других частях системы.

Монолитная архитектура может быть целесообразным выбором для начального этапа разработки программной системы прогнозирования появления новых технологических областей на основе патентной информации. Она позволяет быстро разработать и развернуть рабочую версию системы, упрощает управление кодом и обеспечивает целостность данных. Однако, по мере роста системы и увеличения объема данных, могут возникнуть проблемы с масштабируемостью и сложностью поддержки. При этом даже на начальных этапах разработки системы могут возникнуть трудности с быстрой сменой фокуса системы.

*Сервис-ориентированная архитектура*

Сервис-ориентированная архитектура (SOA) представляет собой подход к разработке программных систем, при котором функциональность приложения разделяется на отдельные, автономные сервисы, которые взаимодействуют друг с другом через стандартизированные интерфейсы. В контексте системы прогнозирования появления новых технологических областей на основе патентной информации, SOA предлагает ряд возможностей для гибкости, масштабируемости и управляемости [51].

На рисунке 10 изображен пример сервис-ориентированной архитектуры.

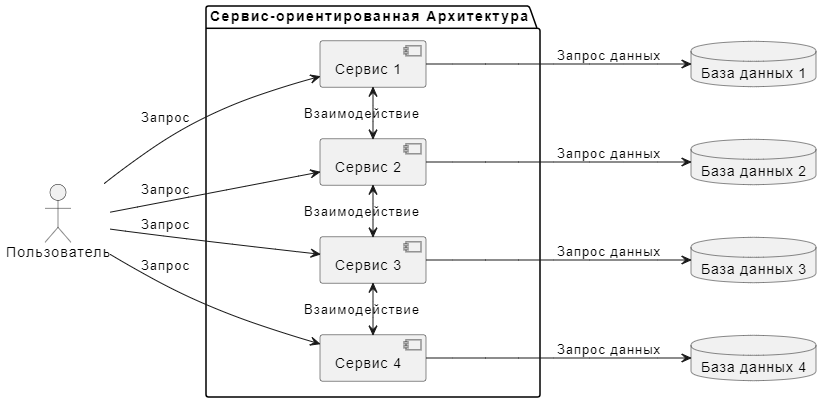


Рисунок 10 – Пример сервис-ориентированной архитектуры

Основные характеристики:

1. декомпозиция на сервисы – в SOA функциональность системы разбивается на независимые сервисы, каждый из которых отвечает за выполнение конкретных задач; например, могут быть отдельные сервисы для парсинга патентных документов, создания и обучения моделей, визуализации данных и генерации отчетов;
2. стандартизированные интерфейсы – сервисы взаимодействуют друг с другом через четко определенные интерфейсы, часто на основе протоколов SOAP или REST; это обеспечивает стандартизированный способ обмена данными и взаимодействия между компонентами;
3. автономность и независимость – каждый сервис является автономным и может быть разработан, развернут и масштабирован независимо от других сервисов; это упрощает управление системой и позволяет легко адаптировать её к изменяющимся требованиям.

Преимущества сервис-ориентированной архитектуры:

1. гибкость и масштабируемость – благодаря разделению функциональности на отдельные сервисы, система может быть легко масштабируема; можно масштабировать только те части системы, которые требуют дополнительных ресурсов, без необходимости масштабировать всё приложение целиком;
2. повышенная надежность – поскольку каждый сервис автономен, сбой в одном из них не приводит к отказу всей системы; это повышает общую надежность и устойчивость приложения;
3. упрощение поддержки и модернизации – независимое развитие и развертывание сервисов упрощает процесс поддержки и модернизации системы; новые функции могут быть добавлены в виде новых сервисов, а существующие сервисы могут быть обновлены без необходимости остановки всей системы;
4. повторное использование – сервисы, разработанные для одной системы, могут быть повторно использованы в других проектах; это способствует более эффективному использованию ресурсов и ускоряет разработку новых приложений.

Недостатки сервис-ориентированной архитектуры:

1. сложность управления – разработка и управление большим количеством независимых сервисов требуют сложной инфраструктуры и эффективной системы оркестрации; это может увеличить затраты на разработку и эксплуатацию системы;
2. повышенные требования к взаимодействию – взаимодействие между сервисами через сети может привести к дополнительным задержкам и снижению производительности; необходимость управления межсервисными коммуникациями может усложнить разработку;
3. безопасность – с увеличением числа точек взаимодействия между сервисами повышаются требования к безопасности системы; необходимо обеспечить надежную аутентификацию и авторизацию для каждого сервиса, а также защиту данных при их передаче.

Сервис-ориентированная архитектура представляет собой подходящий выбор для системы прогнозирования появления новых технологических областей на основе патентной информации, особенно если ожидается значительное расширение функциональности и объемов обрабатываемых данных. Разделение на сервисы позволит гибко адаптировать систему к изменяющимся требованиям, обеспечивая независимое развитие и развертывание отдельных компонентов.

*Микросервисная архитектура*

Микросервисная архитектура (MSA) представляет собой современный подход к разработке программного обеспечения, при котором приложение строится как набор небольших, независимых сервисов, каждый из которых выполняет строго определенную функцию. В контексте системы прогнозирования появления новых технологических областей на основе патентной информации, микросервисная архитектура предлагает значительные преимущества в плане гибкости, масштабируемости и управляемости по сравнению с монолитной и сервис-ориентированной архитектурами [52].

На рисунке 11 изображен пример микросервисной архитектуры.

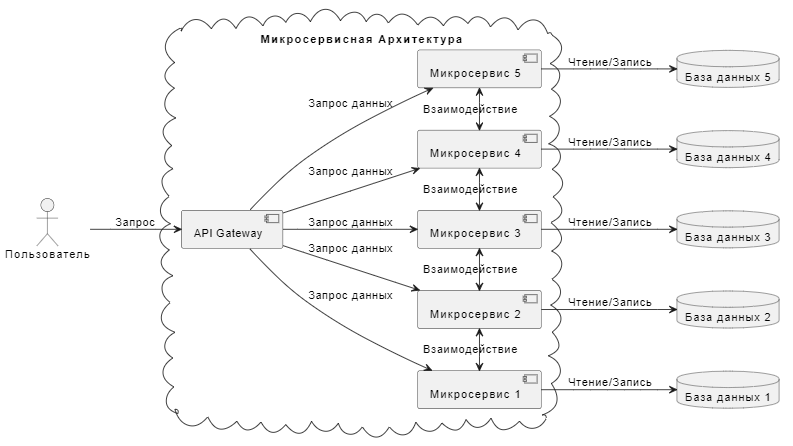


Рисунок 11 – Пример микросервисной архитектуры

Основные характеристики:

1. мелкозернистая декомпозиция – микросервисная архитектура предполагает разделение функциональности на мельчайшие независимые сервисы; каждый микросервис отвечает за выполнение одной задачи или группы тесно связанных задач; например, могут быть отдельные микросервисы для парсинга патентных документов, векторизации текстов, кластеризации данных, визуализации и генерации отчетов;
2. автономность и изоляция – каждый микросервис является полностью автономным и может быть разработан, развернут и масштабирован независимо от других; микросервисы взаимодействуют друг с другом через легковесные протоколы, такие как HTTP/HTTPS, или с помощью очередей сообщений;
3. децентрализованное управление данными – в микросервисной архитектуре каждый сервис может иметь свою собственную базу данных, что позволяет выбирать оптимальные технологии хранения данных для каждой конкретной задачи.

Преимущества микросервисной архитектуры:

1. гибкость разработки и развертывания – разработка микросервисов может выполняться параллельно разными командами, что ускоряет процесс создания системы; микросервисы могут быть развернуты и обновлены независимо друг от друга, что уменьшает время простоя и снижает риск возникновения ошибок;
2. масштабируемость – микросервисы могут быть масштабированы независимо в зависимости от нагрузки; это позволяет эффективно использовать ресурсы и обеспечивать высокую производительность системы даже при увеличении объемов данных;
3. устойчивость к сбоям – сбои в одном микросервисе не приводят к отказу всей системы; это повышает общую надежность приложения, так как проблемы могут быть изолированы и устранены без влияния на другие части системы;
4. технологическая гибкость – разные микросервисы могут быть разработаны с использованием различных технологий и языков программирования, что позволяет выбирать наиболее подходящие инструменты для каждой конкретной задачи;
5. упрощенная поддержка и модернизация – микросервисы легко поддерживать и модернизировать, так как изменения в одном сервисе не требуют изменений в других; это снижает сложность системы и облегчает добавление новых функций.

Недостатки микросервисной архитектуры:

1. сложность управления – управление множеством микросервисов требует развитой системы оркестрации и мониторинга; это может увеличить сложность инфраструктуры и потребовать дополнительных ресурсов для поддержания стабильности системы;
2. повышенные требования к взаимодействию – взаимодействие между микросервисами через сеть может привести к увеличению задержек и повышению затрат на сетевые ресурсы; необходимо тщательно проектировать межсервисные коммуникации для обеспечения высокой производительности.

Микросервисная архитектура является отличным выбором для системы прогнозирования появления новых технологических областей на основе патентной информации. Разделение на микросервисы позволит гибко адаптировать систему к изменяющимся требованиям, обеспечивая независимое развитие и развертывание отдельных компонентов.

Каждый из этих микросервисов может быть разработан, развернут и масштабирован независимо, что обеспечит высокую производительность и устойчивость системы. Микросервисная архитектура также позволит интегрировать новые функции и технологии по мере необходимости, что сделает систему более адаптивной и инновационной.

*Выбор архитектуры*

После детального анализа различных архитектурных подходов предлагается выбрать гибридную микросервисную архитектуру для реализации системы прогнозирования появления новых технологических областей на основе патентной информации. Этот выбор обусловлен необходимостью сочетания преимуществ микросервисной архитектуры с некоторыми аспектами монолитного подхода для обеспечения более простой согласованности данных.

В рамках гибридной микросервисной архитектуры система будет разбита на мелкие, автономные сервисы, каждый из которых будет выполнять конкретные задачи. Например, отдельные микросервисы будут отвечать за парсинг патентных документов, создание и обучение моделей, визуализацию данных и генерацию отчетов. Это обеспечит независимость разработки, развертывания и масштабирования каждого сервиса, что значительно упростит управление системой и повысит ее гибкость и устойчивость.

Для обеспечения более простой согласованности данных и упрощения взаимодействия между сервисами будет использоваться единая база данных. Это решение позволит избежать сложностей, связанных с синхронизацией данных между разными базами данных, и упростит транзакционное управление. Единая база данных обеспечит целостность и непротиворечивость данных, что особенно важно для систем, работающих с критически важной информацией, такой как патентные данные.

Гибридный подход позволяет использовать сильные стороны микросервисной архитектуры, такие как высокая гибкость, масштабируемость и независимость компонентов, при этом сохраняя простоту и целостность данных, характерные для монолитных систем. Это оптимальное решение для создания надежной и адаптивной системы прогнозирования, способной эффективно обрабатывать большие объемы данных и адаптироваться к изменяющимся требованиям.

Таким образом, выбор гибридной микросервисной архитектуры позволяет достичь баланса между гибкостью и управляемостью, обеспечивая надежную основу для долгосрочного развития и масштабирования системы прогнозирования появления новых технологических областей на основе патентной информации.

### 2.1.2 Структурная схема программной системы

Структурная система программной системы прогнозирования новых технологических областей включает несколько модулей, каждый из которых выполняет определенные функции. Это обеспечивает высокую гибкость, масштабируемость и модульность системы, позволяя эффективно управлять данными и упрощая сопровождение и модернизацию.

Основными модулями системы являются:

1. модуль построения моделей – отвечает за создание и обучение моделей на основе патентных данных; включает в себя алгоритмы преобразования текстов патентов в векторные представления с использованием модели SentenceTransformer и методы кластеризации данных, такие как DBSCAN; модуль построения моделей получает данные из модуля хранения данных, обрабатывает их и сохраняет результаты обратно в базу данных для последующего использования другими модулями;
2. модуль пользовательского интерфейса – обеспечивает взаимодействие пользователя с системой; включает графический интерфейс для загрузки патентных документов, настройки фильтров и параметров моделей, а также для просмотра результатов анализа и прогнозов; модуль пользовательского интерфейса обменивается данными с модулем хранения данных и другими модулями для выполнения запросов пользователя и отображения результатов;
3. модуль визуализации – отвечает за визуализацию данных и результатов анализа; включает генерацию патентных ландшафтов с использованием UMAP и LDA, а также создание различных графиков и диаграмм для представления информации в удобной для пользователя форме; данный модуль получает данные и результаты анализа из модуля хранения данных и модуля построения моделей, обрабатывает их и отображает в пользовательском интерфейсе;
4. модуль генерации отчетов – отвечает за создание отчетов на основе анализа данных и прогнозов; включает генерацию гипотез и оценку их вероятностей, а также формирование отчетов в читаемом виде для пользователей; модуль генерации отчетов получает данные и результаты анализа из модуля хранения данных и других модулей для генерации финальных отчетов;
5. модуль хранения данных – является центральным хранилищем для всех данных системы, включая документы с патентами, векторные представления, модели, прогнозы и параметры; обеспечивает целостность и непротиворечивость данных; он обеспечивает хранение и доступ к данным для всех остальных модулей системы, обеспечивая единое место для управления данными.

Модули системы взаимосвязаны и обмениваются данными через модуль хранения данных. Пользователь взаимодействует с системой через модуль пользовательского интерфейса, который отправляет запросы другим модулям для выполнения различных задач, таких как построение моделей, визуализация данных и генерация отчетов. Модули построения моделей, визуализации и генерации отчетов получают необходимые данные из модуля хранения данных и возвращают результаты обратно, обеспечивая целостность и согласованность информации в системе. Структурная схема архитектуры программной системы изображена на рисунке 12.

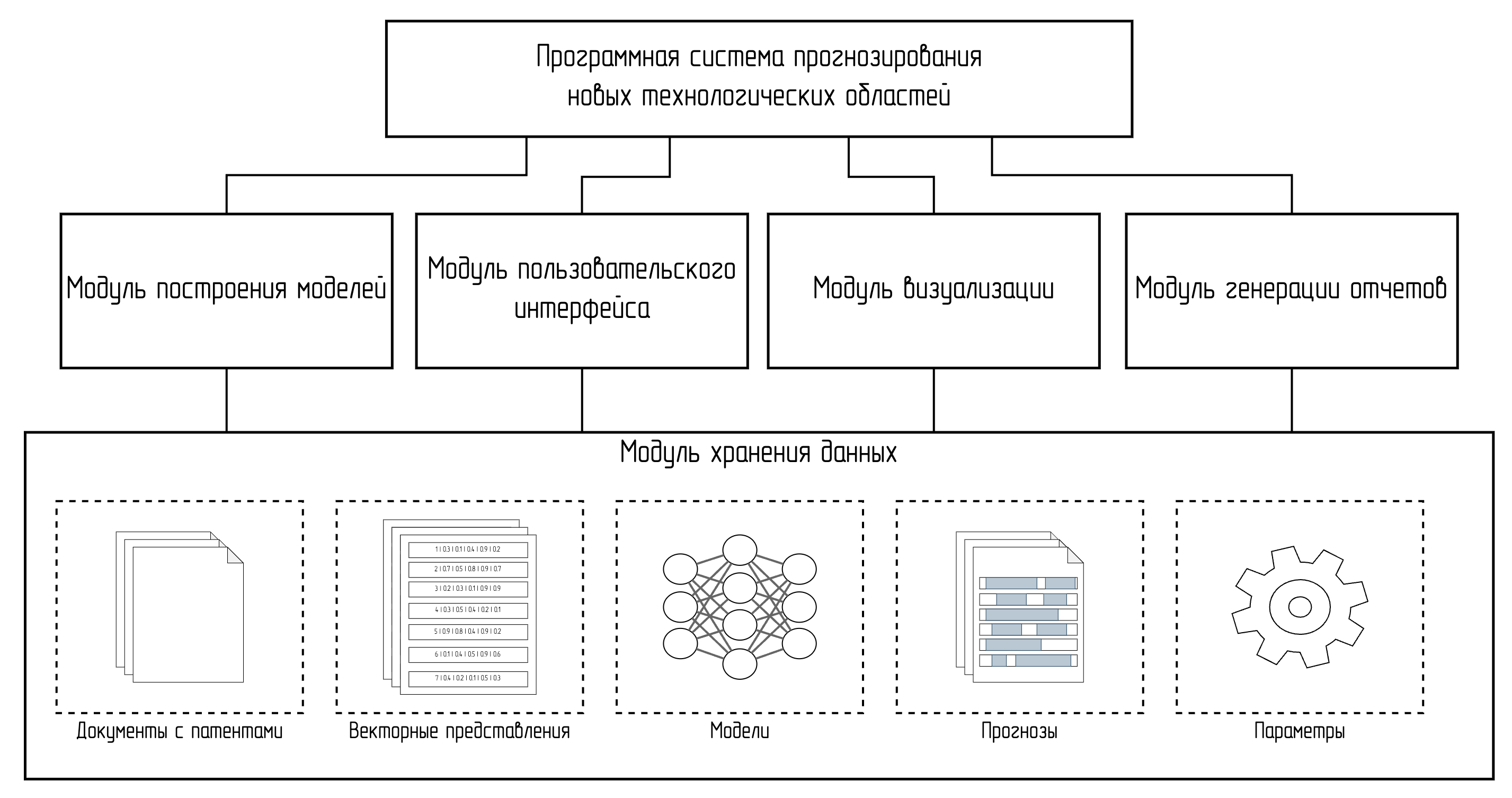


Рисунок 12 – Схема структурная программной системы

Такой подход обеспечивает высокую модульность и гибкость системы, позволяя легко добавлять новые функции и адаптировать существующие под изменяющиеся требования и объемы данных.

## 2.2 Выбор средств разработки

В основе разработки данной системы выбран язык программирования Python [53]. Это обусловлено его возможностями для обработки данных, машинного обучения и анализа информации. Python обладает обширным набором библиотек и инструментов, таких как pandas, NumPy, scikit-learn, TensorFlow и многие другие, которые позволяют эффективно реализовать сложные алгоритмы и модели. Кроме того, Python является высокоуровневым и удобным для разработки, что ускоряет процесс создания и отладки программного обеспечения.

Для разработки пользовательского интерфейса выбран JavaScript, который является стандартом для веб-разработки [54]. С использованием современных фреймворков, таких как React или Vue.js [55], можно создать интерактивный и отзывчивый интерфейс, обеспечивающий удобное взаимодействие пользователя с системой.

В последующих главах будут рассмотрены выбор и использование конкретных средств разработки для различных аспектов системы.

### 2.2.1 Машинное обучение

Для реализации машинного обучения в программной системе прогнозирования новых технологических областей на основе патентной информации будут использованы проверенные инструменты и библиотеки Python. В основе системы лежит модель SentenceTransformer [56], которая позволяет эффективно преобразовывать текстовые данные патентов в векторные представления. Эта модель основана на методах глубокого обучения и использует архитектуры трансформеров, такие как BERT и его производные, что обеспечивает высокое качество векторных представлений и, следовательно, улучшает результаты последующего анализа и кластеризации.

Для выполнения кластеризации данных будет использоваться метод DBSCAN, который доступен в библиотеке scikit-learn [57]. DBSCAN является плотностным алгоритмом кластеризации, который хорошо справляется с выявлением кластеров произвольной формы и устойчив к шуму в данных. Это делает его идеальным выбором для анализа патентных данных, где структуры могут быть сложными и разрозненными. Важно отметить, что для оптимизации параметра eps в DBSCAN будет применяться техника градиентного спуска, что позволит найти оптимальное значение и улучшить метрику кластеризации, такую как Adjusted Rand Index.

Кроме того, для оценки качества кластеризации и ассоциации меток кластеров с метками классов IPC будут активно использоваться метрики из scikit-learn, включая Adjusted Rand Index. Эти инструменты позволят объективно оценивать результаты кластеризации и обеспечат надежную ассоциацию кластеров с технологическими классами.

Использование этих инструментов и библиотек обеспечит высокую точность, гибкость и производительность системы, позволяя эффективно анализировать большие объемы патентных данных и прогнозировать появление новых технологических областей.

### 2.2.2 Визуализация

Основным методом для создания визуальных представлений данных будет использоваться UMAP (Uniform Manifold Approximation and Projection). UMAP – это метод нелинейного уменьшения размерности, который превосходно подходит для визуализации высоко размерных данных, таких как векторные представления патентных текстов [58]. Этот метод сохраняет структурные особенности данных и позволяет создать двух- или трехмерные представления, которые можно легко интерпретировать. Преимущество UMAP заключается в его способности сохранять глобальную структуру данных, что особенно важно для выявления скрытых паттернов и кластеров в патентных данных.

Для генерации ключевых слов и тем, связанных с различными кластерами, будет использован метод LDA (Latent Dirichlet Allocation), доступный в библиотеке Gensim [59]. LDA позволяет выявлять скрытые темы в текстовых данных и автоматически генерировать ключевые слова для каждого кластера. Эти ключевые слова помогут пользователям быстро понять основное содержание и тематическую направленность каждого кластера, улучшая интерпретируемость результатов кластеризации.

Визуализация результатов будет осуществляться с помощью библиотек matplotlib и Plotly [60]. Matplotlib предоставляет широкий спектр возможностей для создания статических графиков и диаграмм, которые можно использовать для базовой визуализации данных и результатов анализа. Plotly, в свою очередь, позволяет создавать интерактивные графики, которые можно легко интегрировать в веб-интерфейс системы. Это обеспечит пользователю возможность взаимодействовать с визуализациями, изменять параметры отображения и получать более детальную информацию по каждому элементу.

Использование UMAP и LDA в сочетании с популярными инструментами визуализации, такими как matplotlib и Plotly, обеспечит создание информативных и наглядных представлений данных. Это позволит пользователям эффективно анализировать патентные данные и принимать обоснованные решения на основе полученной информации.

### 2.2.3 Хранение данных

Для эффективного хранения данных в программной системе прогнозирования новых технологических областей на основе патентной информации будут использованы два компонента: MinIO [61] и PostgreSQL [62]. Это сочетание обеспечит надежное хранение и быстрый доступ к различным типам данных, необходимым для работы системы.

Файлы патентных документов будут загружаться и храниться в MinIO. MinIO – это высокопроизводительное объектное хранилище, совместимое с API Amazon S3. Оно обеспечивает надежное и масштабируемое хранение файлов любых размеров, что идеально подходит для хранения большого объема патентных документов. MinIO позволяет эффективно управлять файлами, обеспечивает высокую доступность и отказоустойчивость, что критически важно для системы, работающей с большими объемами данных. Использование MinIO также упрощает интеграцию с другими компонентами системы через стандартные API, облегчая процесс загрузки и извлечения файлов.

Структурированные данные, извлеченные из патентных документов, будут храниться в реляционной базе данных PostgreSQL. PostgreSQL – это открытая система управления базами данных, которая поддерживает широкий спектр функций, необходимых для хранения и управления данными. Она обеспечивает высокую производительность, надежность и возможность масштабирования, что делает её отличным выбором для систем, работающих с большими объемами данных и требующих сложных запросов.

Данные будут разделены на несколько таблиц, таких как патенты, векторные представления, модели, прогнозы и параметры. Это позволит организовать данные в логически связанные структуры, обеспечивая быстрый доступ и эффективное управление. PostgreSQL также поддерживает сложные транзакции и обеспечивает целостность данных, что особенно важно для работы с критически важной информацией, такой как патентные данные.

Использование PostgreSQL позволит эффективно выполнять сложные запросы для анализа данных, построения моделей и генерации отчетов. Это обеспечит высокую производительность и надежность системы, позволяя пользователям быстро получать доступ к необходимой информации и результатам анализа.

Таким образом, сочетание MinIO и PostgreSQL обеспечит надежное и эффективное хранение как файловых, так и структурированных данных. MinIO будет использоваться для хранения патентных документов, обеспечивая их доступность и масштабируемость, а PostgreSQL – для управления структурированными данными, обеспечивая высокую производительность и целостность информации.

### 2.2.4 Клиент-серверное взаимодействие

Основной протокол взаимодействия между клиентом и сервером будет REST [63], реализованный с помощью FastAPI [64]. Для оптимизации передачи данных и уменьшения частоты запросов будет также внедрен механизм отправки событий через WebSocket [65] с использованием Centrifugo [66].

Пример временной диаграммы взаимодействия клиента и серверной части представлен на рисунке 13.

FastAPI – это современный веб-фреймворк для создания API на языке Python, обеспечивающий высокую производительность, поддержку асинхронного программирования и простоту разработки. В системе FastAPI будет использоваться для реализации REST API, который обеспечит взаимодействие между клиентской и серверной частями. REST API предоставит стандартные HTTP-методы для выполнения основных операций, таких как загрузка патентных документов, настройка фильтров, запуск моделирования и получение результатов анализа. Основные преимущества использования FastAPI включают высокую производительность, простоту разработки и поддержку автоматической генерации OpenAPI документации [67], что упрощает интеграцию и тестирование API.

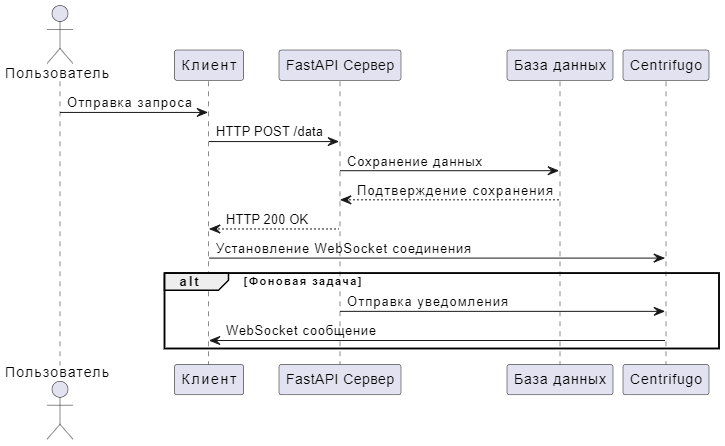


Рисунок 13 – Пример клиент-серверного взаимодействия системы

Для обеспечения реального времени и уменьшения частоты запросов на сервер будет использоваться механизм отправки событий через WebSocket. В этом случае выбран Centrifugo, сервер для работы с WebSocket, который поддерживает масштабируемые и надежные соединения. Centrifugo обеспечит отправку событий в реальном времени, позволяя пользователям получать обновления без необходимости постоянно опрашивать сервер, и легко масштабируется, что обеспечивает надежную работу системы при увеличении числа пользователей. Он также гибко интегрируется с FastAPI и другими компонентами системы, обеспечивая эффективную архитектуру.

Взаимодействие между клиентом и сервером будет происходить следующим образом: клиент отправляет запросы на сервер для выполнения операций, таких как загрузка данных, запуск анализа и получение результатов, сервер обрабатывает запросы с использованием FastAPI и возвращает ответы в формате JSON. Для получения обновлений в реальном времени клиент устанавливает соединение с сервером Centrifugo через WebSocket, и сервер отправляет события, такие как завершение построения модели или генерация отчета, непосредственно на клиент, обеспечивая мгновенную обратную связь. Такое сочетание REST API с использованием FastAPI и механизма WebSocket с использованием Centrifugo обеспечит эффективное и масштабируемое клиент-серверное взаимодействие, предоставляя пользователям удобный и отзывчивый интерфейс для работы с системой прогнозирования новых технологических областей на основе патентной информации.

### 2.2.5 Инфраструктура

Для обеспечения надежной и масштабируемой инфраструктуры программный продукт будет реализован с использованием системы контейнеризации Docker [68]. Docker предоставляет инструменты для создания, развертывания и управления контейнерами, что позволяет значительно упростить процесс разработки, тестирования и эксплуатации программного обеспечения.

Контейнеризация с помощью Docker позволяет изолировать все компоненты системы, включая базу данных, серверные и клиентские приложения, а также вспомогательные сервисы. Это обеспечивает консистентность окружения на всех этапах жизненного цикла разработки, от локальной среды разработчика до производственных серверов. Благодаря этому подходу можно гарантировать, что код, протестированный в одном окружении, будет работать идентично в любом другом окружении, что существенно снижает вероятность возникновения ошибок при переносе между различными системами.

Docker позволяет упаковать все зависимости и конфигурации приложения в контейнеры, что облегчает процесс развертывания и масштабирования системы. Например, отдельные контейнеры могут быть созданы для FastAPI сервера, базы данных PostgreSQL, сервера MinIO для хранения патентных документов и Centrifugo для обеспечения WebSocket соединений. Это позволяет легко управлять и обновлять каждый компонент системы независимо от других, обеспечивая гибкость и модульность инфраструктуры.

Использование Docker также упрощает процесс горизонтального масштабирования. При увеличении нагрузки на систему можно легко создать дополнительные экземпляры нужных контейнеров и распределить нагрузку с помощью оркестраторов контейнеров, таких как Kubernetes. Это обеспечивает высокую доступность и устойчивость системы к отказам, что особенно важно для критически важных приложений, работающих с большими объемами данных.

Таким образом, реализация программного продукта через систему контейнеризации Docker обеспечит надежную, гибкую и масштабируемую инфраструктуру, способную эффективно поддерживать все компоненты системы прогнозирования новых технологических областей на основе патентной информации.

## 2.3 Разработка метода прогнозирования

### 2.3.1 Метод построения модели

Метод построения модели в системе прогнозирования новых технологических областей на основе патентной информации включает несколько этапов, которые обеспечивают преобразование текстовых данных в векторные представления, кластеризацию и оптимизацию параметров. Данный метод позволяет создать высокоэффективные модели, способные выявлять скрытые паттерны в патентных данных и прогнозировать появление новых технологических областей.

На первом этапе осуществляется преобразование текстов патентов (Title и Novelty) в векторные представления с использованием модели SentenceTransformer. SentenceTransformer основан на современных архитектурах трансформеров, таких как BERT, и позволяет получить высококачественные векторные представления текстов. Эти вектора хранятся в базе данных, что позволяет повторно использовать их без необходимости пересчета.

После получения векторных представлений для каждого патента запускается процесс кластеризации методом DBSCAN, который доступен в библиотеке scikit-learn. DBSCAN (Density-Based Spatial Clustering of Applications with Noise) является плотностным алгоритмом, который хорошо подходит для выявления кластеров произвольной формы и устойчив к шуму в данных. Он позволяет выделить группы патентов, которые имеют схожие тематические и технологические характеристики.

Для оценки качества кластеризации используется метрика Adjusted Rand Index (ARI), которая позволяет измерить степень соответствия полученных кластеров с известными метками классов IPC. ARI учитывает случайное совпадение и предоставляет более точную оценку, чем простые метрики. На данном этапе осуществляется анализ полученных кластеров и их соответствие классификации по меткам IPC.

С целью повышения качества кластеризации и улучшения метрики ARI проводится оптимизация параметра в алгоритме DBSCAN. Для этого используется метод градиентного спуска, который позволяет найти оптимальное значение параметра , максимизирующее метрику ARI. Оптимизация осуществляется итеративно, что позволяет добиться наилучшего разделения патентов на кластеры.

После завершения кластеризации и оптимизации параметров осуществляется ассоциация меток кластеров с метками классов IPC. Этот процесс включает попарное сравнение точности кластеризации и меток IPC, что позволяет определить, насколько хорошо кластеры повторяют классификацию. При высоком соответствии метки кластеров могут быть ассоциированы с метками классов IPC, что упрощает дальнейший анализ и интерпретацию данных. Структурная схема алгоритма ассоциации представлена на рисунке 14.

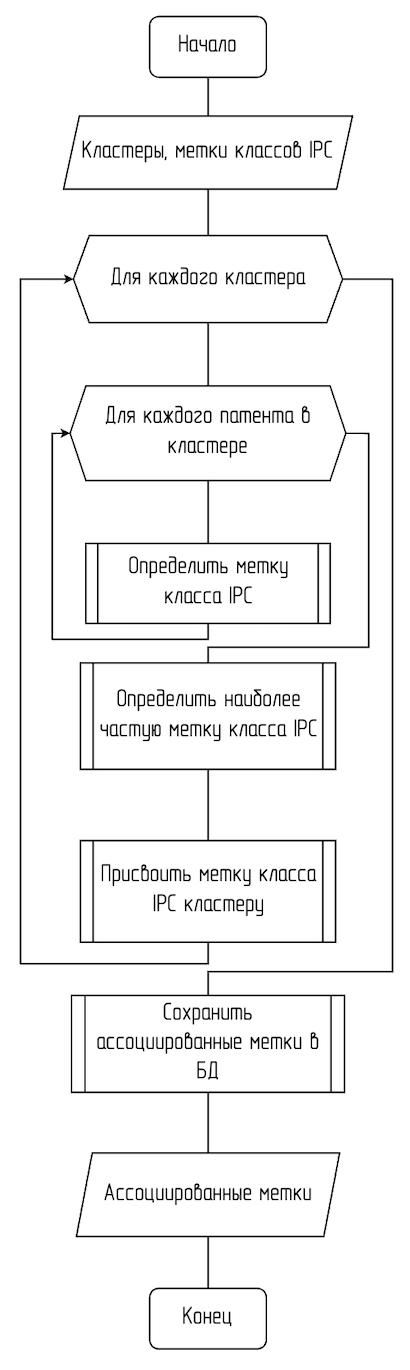


Рисунок 14 – Алгоритм ассоциации меток классов с кластерами

После завершения всех этапов модель сохраняется в базе данных. Включаются порог кластеризации, метрика ARI, метки кластеров и ассоциированные с ними метки классов IPC. Это позволяет повторно использовать модель для анализа новых данных и оценки её качества.

На рисунке 15 представлена структурная схема алгоритма построения модели и оптимизации параметров кластеризации. Эта схема визуализирует последовательность выполнения основных этапов метода и взаимосвязь между ними, что способствует лучшему пониманию и реализации процесса построения модели.

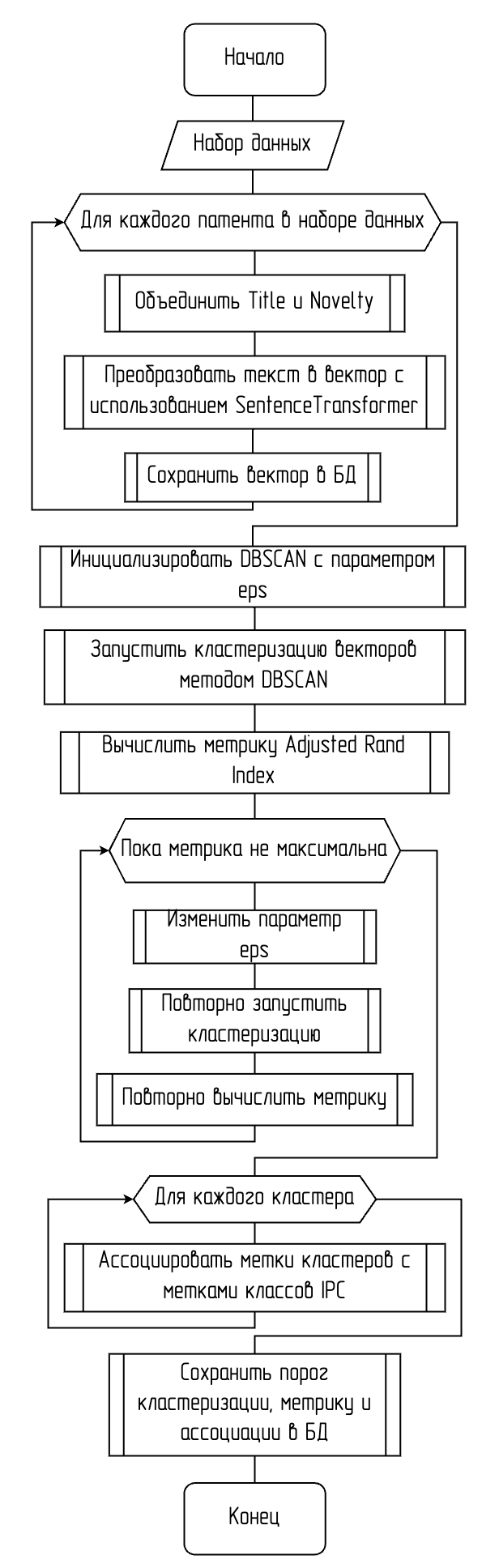


Рисунок 15 – Алгоритм построения модели патентного ландшафта

### 2.3.2 Метод построения прогнозной модели

Метод построения прогнозной модели в системе прогнозирования новых технологических областей на основе патентной информации опирается на аналогичные принципы, что и метод построения базовой модели, однако включает дополнительные этапы для учета временных аспектов и генерации прогнозов. Этот метод позволяет определить возможные направления развития технологий и выявить потенциальные новые технологические области на основе анализа изменений в патентных данных за определенные временные периоды.

Первым шагом является создание нового набора данных, аналогичного тому, который использовался для построения базовой модели, но охватывающего более поздний временной диапазон. Этот новый набор данных фильтруется по тем же критериям, что и исходный, чтобы обеспечить сопоставимость результатов. Важно, чтобы коды IPC и другие параметры фильтрации оставались идентичными для обоих наборов данных.

Как и в методе построения базовой модели, текстовые данные патентов из нового набора данных преобразуются в векторные представления с использованием модели SentenceTransformer. Полученные вектора хранятся в базе данных для последующего использования в процессе кластеризации и анализа.

Для кластеризации данных в новом наборе используется метод DBSCAN с оптимизированным параметром , найденным на этапе построения базовой модели. Это обеспечивает согласованность и позволяет сравнивать результаты кластеризации для разных временных периодов. Метод DBSCAN выделяет группы патентов, которые имеют схожие тематические и технологические характеристики, что позволяет выявлять изменения в патентном ландшафте.

После кластеризации проводится оценка качества кластеризации с использованием метрики Adjusted Rand Index (ARI), аналогично базовой модели. Далее осуществляется ассоциация меток кластеров с метками классов IPC, присутствующими в фильтре нового набора данных. Это позволяет определить, насколько хорошо кластеры отражают классификацию патентов по классам IPC, и выявить изменения в тематических областях.

На основе результатов кластеризации и ассоциации меток создается прогнозный патентный ландшафт. Для визуализации изменений в патентных данных используется метод UMAP, который позволяет построить двух- или трехмерное представление кластеров и их взаимосвязей. Этот ландшафт наглядно показывает, как изменились технологические области и где могут появиться новые кластеры.

После завершения всех этапов прогнозная модель сохраняется в базе данных. Включаются порог кластеризации, метрика ARI, метки кластеров и ассоциированные с ними метки классов IPC. Это позволяет повторно использовать модель для анализа новых данных и оценки её качества.

На рисунке 16 представлены структурные схемы алгоритмов построения прогнозной модели. Эти схемы визуализируют последовательность выполнения основных этапов метода и взаимосвязь между ними, что способствует лучшему пониманию и реализации процесса прогнозирования.

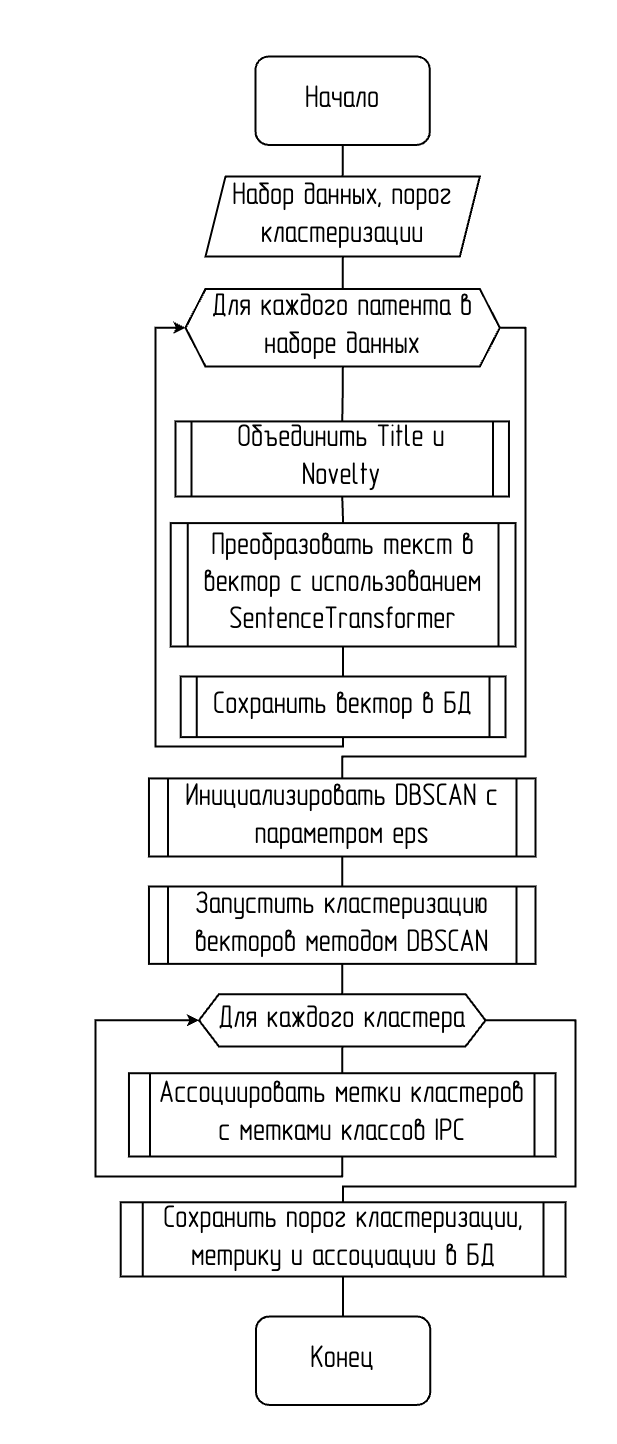


Рисунок 16 – Алгоритм построения прогнозной модели патентного ландшафта

### 2.3.3 Метод генерации отчета

Метод генерации отчета в системе прогнозирования новых технологических областей основывается на анализе изменений в патентных данных и выявлении значимых паттернов с использованием матрицы ассоциаций. Этот процесс позволяет формировать гипотезы о возможных направлениях развития технологий и оценивать вероятность их возникновения.

На начальном этапе создается матрица ассоциаций, в которой по оси X представлены кластеры базовой модели, а по оси Y – кластеры прогнозной модели. В каждой ячейке матрицы содержится процент соответствия между кластерами. Этот показатель вычисляется как отношение числа патентов, присутствующих в обоих кластерах, к общему числу патентов в базовом кластере, что позволяет оценить изменения в патентном ландшафте и выявить значимые паттерны.

Следующим шагом является инициализация списка гипотез, который будет заполняться на основе анализа матрицы ассоциаций. Этот список будет содержать возможные сценарии развития.

Для каждой строки в матрице ассоциаций определяется класс модели (X-кластер). Далее для этой строки находятся ячейки с соответствием выше заданного порога. Количество таких ячеек определяет дальнейший ход анализа:

* если ячеек больше одной, проводится проверка, принадлежат ли они к различным классам (Y-кластерам); если да, создается гипотеза о слиянии классов и вероятности появления новой технологической области; в противном случае создается гипотеза о диверсификации и смене фокуса внутри класса;
* если ячейка одна, создается гипотеза о сохранении фокуса технологической области;
* если ячеек нет и общее соответствие меньше порога, создается гипотеза о снижении активности в данной технологической области.

После формирования гипотез проводится оценка их вероятности. Эта оценка основывается на анализе степени изменений в матрице ассоциаций, количестве патентов и других показателях. Для каждого типа гипотезы используется своя формула для вычисления вероятности:

1. гипотеза о слиянии классов – вероятность рассчитывается на основе суммы процентного соответствия всех ячеек, участвующих в слиянии, и средней темпа роста патентов в этих кластерах:

где – процентное соответствие для каждой ячейки, n – число ячеек, G – средний тем роста патентов;

1. гипотеза о диверсификации и смене фокуса – вероятность основывается на проценте патентов, переместившихся в новый кластер, и изменении тематики ключевых слов, выявленных с помощью LDA:

где – число патентов, переместившихся в новый кластер, – общее число патентов в исходном кластер, – коэффициент изменения ключевых слов;

1. гипотеза о снижении активности – вероятность оценивается на основе общего снижения активности в кластере, вычисляемого как разница в количестве патентов между базовым и прогнозным периодами:

где – число патентов в прогнозном периоде, – число патентов в базовом периоде.

На завершающем этапе гипотезы и их вероятности сохраняются в базе данных. Это позволяет пользователям системы получать доступ к проанализированным данным и принимать обоснованные решения на основе полученных прогнозов.

На рисунке 17 представлена структурная схема алгоритма генерации отчетов. Эта схема визуализирует процесс анализа матрицы ассоциаций и формирования гипотез, показывая последовательность выполнения основных этапов и условия перехода между ними.

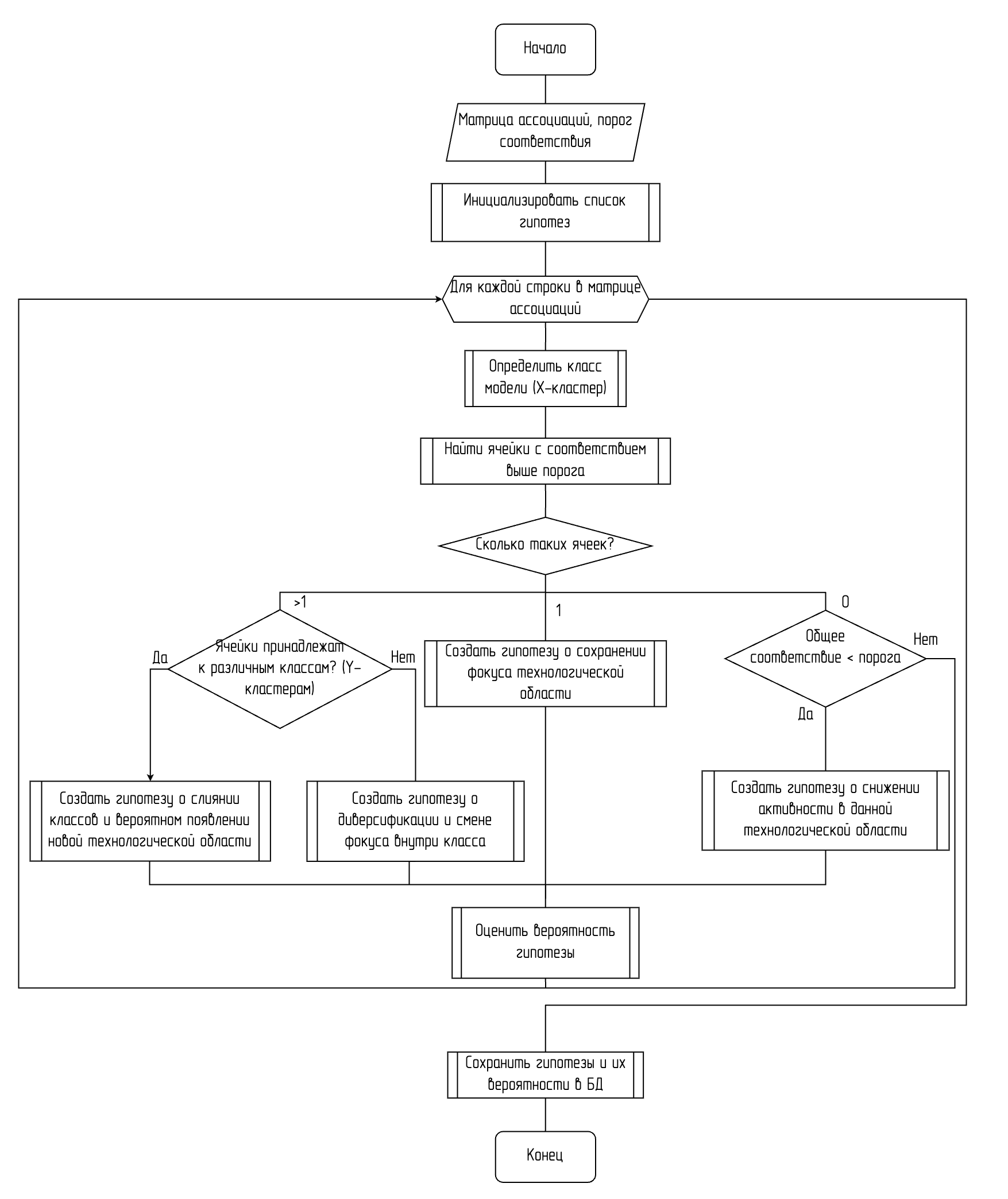


Рисунок 17 – Алгоритм генерации отчета

Метод генерации отчета является главным элементом системы прогнозирования, обеспечивающим выявление значимых паттернов и формирование гипотез о возможных направлениях развития технологий. Использование этого метода позволяет эффективно анализировать большие объемы данных и предоставлять пользователям ценные инсайты для принятия стратегических решений в сфере инноваций и технологий.

## 2.4 Разработка концептуальной модели предметной области

Концептуальная модель описывает основные сущности, их свойства и взаимосвязи, что позволяет более четко понять структуру данных и процессы, происходящие в системе. На рисунке 18 изображена концептуальная модель предметной области, которая включает в себя все основные компоненты системы и их взаимосвязи.

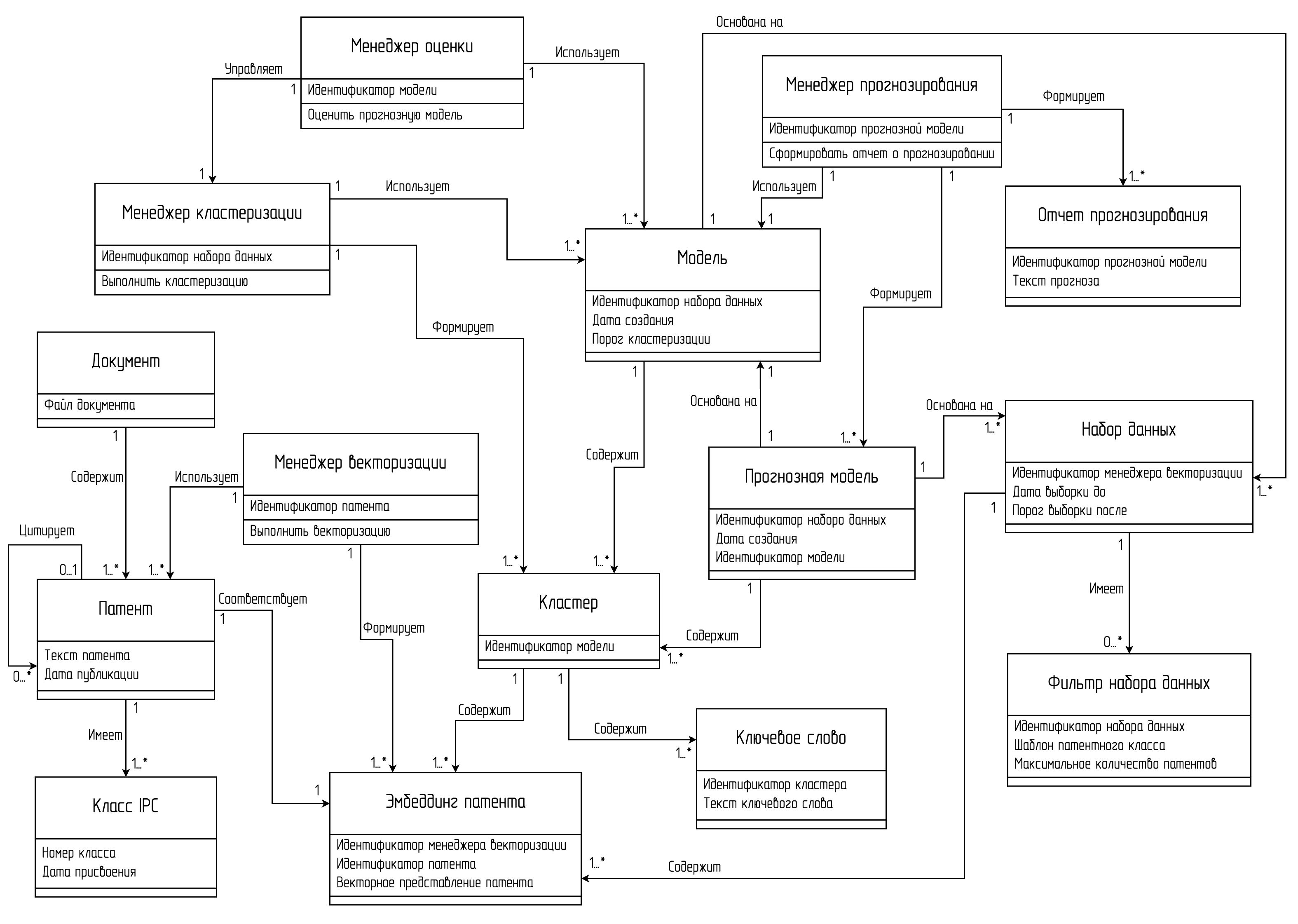


Рисунок 18 – Концептуальная модель предметной области

Документ представляет собой файл, содержащий патентную информацию. Каждый документ может содержать множество патентов, причем каждый патент может цитировать другие патенты. Патент включает в себя текст патента и дату публикации, и он может быть классифицирован по одному или нескольким классам Международной патентной классификации (IPC). Классы IPC характеризуются номером класса и датой присвоения, и каждый класс может соответствовать множеству патентов.

Эмбеддинги патентов формируются менеджером векторизации и включают идентификатор менеджера векторизации, идентификатор патента и его векторное представление. Эти эмбеддинги соответствуют конкретным патентам и используются для дальнейшей кластеризации. Кластер, в свою очередь, ассоциируется с моделью и содержит множество эмбеддингов патентов. Каждый кластер может иметь ключевые слова, которые описывают его содержимое. Ключевые слова характеризуются текстом и идентификатором кластера, которому они принадлежат.

Модель представляет собой результат кластеризации патентных данных и включает идентификатор набора данных, дату создания и порог кластеризации. Модель используется для построения прогнозной модели, которая основана на аналогичных данных, но охватывает более поздний временной диапазон. Прогнозная модель также включает идентификатор набора данных, дату создания и идентификатор модели, на основе которой она была создана.

Менеджер кластеризации отвечает за выполнение кластеризации патентных данных и использует идентификатор набора данных для идентификации данных, на которых будет проведена кластеризация. Менеджер векторизации выполняет преобразование текстовых данных патентов в векторные представления. Менеджер оценки отвечает за оценку прогнозных моделей, используя идентификатор модели для идентификации данных, которые необходимо оценить. Менеджер прогнозирования формирует отчет о прогнозировании на основе идентификатора прогнозной модели, оценивая ее результаты и создавая гипотезы о возможных направлениях развития технологий.

Набор данных включает идентификатор менеджера векторизации, даты выбора данных до и после определенного момента, и фильтры набора данных, которые определяют критерии выбора патентов. Фильтр набора данных включает шаблон патентного класса и максимальное количество патентов, которые могут быть выбраны.

Отчет прогнозирования, который формируется на основе данных прогнозной модели, включает идентификатор прогнозной модели и текст прогноза, представляя результаты анализа в понятной и структурированной форме.

Таким образом, концептуальная модель предметной области охватывает все основные аспекты системы прогнозирования новых технологических областей и обеспечивает четкое понимание структуры данных и процессов, что способствует эффективному проектированию и реализации системы.

## 2.5. Разработка программного интерфейса

Разработка программного интерфейса является главным аспектом создания системы прогнозирования новых технологических областей, обеспечивая взаимодействие пользователя с функциональностью системы. На рисунке 19 изображена диаграмма вариантов использования, которая иллюстрирует основные действия пользователя и соответствующие им функции системы.

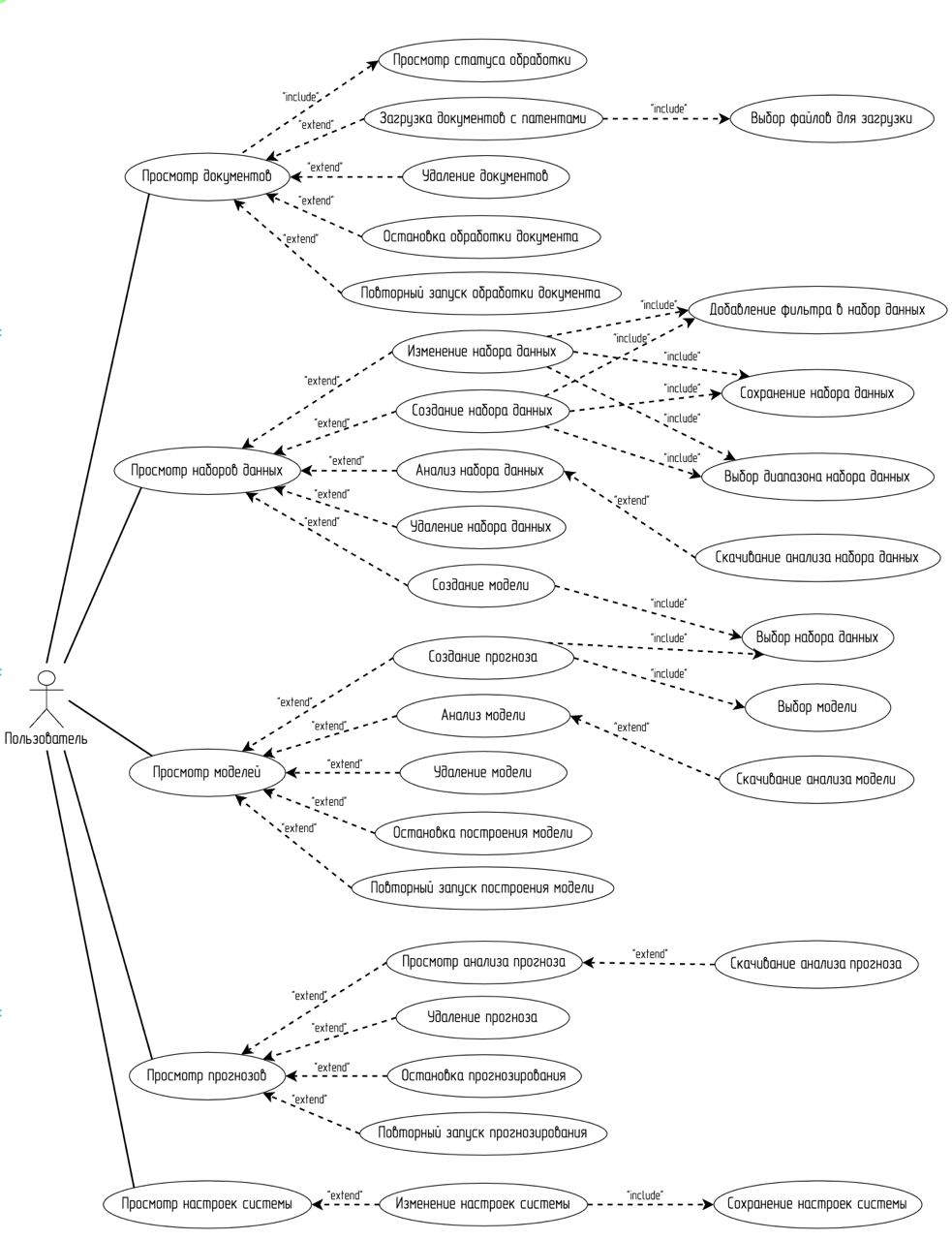


Рисунок 19 – Диаграмма вариантов использования

Варианты использования включают:

* просмотр документов включает в себя действия, связанные с управлением патентными документами; пользователь может загружать документы, выбирая файлы для загрузки, просматривать статус обработки загруженных документов, удалять документы и останавливать их обработку; в случае необходимости возможен повторный запуск обработки документа;
* просмотр наборов данных охватывает действия, связанные с управлением наборами данных, используемыми для анализа и моделирования; пользователь может создавать и изменять наборы данных, добавляя фильтры и сохраняя наборы; возможен анализ набора данных, включающий выбор диапазона и скачивание результатов анализа; также пользователь может удалять наборы данных;
* просмотр моделей включает функции, связанные с созданием и управлением моделями; пользователь может создавать модели, выбирая соответствующие наборы данных, и анализировать их; также доступны действия по удалению моделей, остановке их построения и повторному запуску процесса построения;
* просмотр прогнозов охватывает действия, связанные с прогнозированием и анализом результатов; пользователь может создавать прогнозы, выбирая соответствующие модели и наборы данных, и анализировать их; доступны функции для удаления прогнозов, остановки процесса прогнозирования и повторного запуска; пользователь может также просматривать анализ прогнозов и скачивать результаты;
* просмотр настроек системы включает действия по изменению и сохранению настроек системы, что позволяет пользователю настраивать параметры системы в соответствии с их потребностями.

Каждое из этих действий может включать дополнительные поддействия, которые расширяют основные функции, такие как добавление фильтров, выбор диапазонов и скачивание результатов анализа. Диаграмма вариантов использования наглядно демонстрирует все возможные взаимодействия пользователя с системой, обеспечивая полное представление о функциональности программного интерфейса.

Для обеспечения всех функций, представленных на диаграмме вариантов использования, необходимо разработать набор конечных точек REST API. Эти конечные точки обеспечат взаимодействие клиентской части системы с серверной, позволяя выполнять все необходимые операции. В качестве фреймворка для создания REST API будет использоваться FastAPI. Конечные точки представлены в таблицах 5 – 9.

Таблица 5 – Конечные точки раздела «Документы»

| **Название** | **Описание** | **Параметры** |
| --- | --- | --- |
| Загрузка документов | Загружает патентные документы в систему | files (body): файлы для загрузки |
| Просмотр статуса обработки | Возвращает статус обработки документа | document\_id (path): ID документа |
| Просмотр документов | Возвращает список всех загруженных документов | - |
| Удаление документа | Удаляет указанный документ из системы | document\_id (path): ID документа |
| Остановка обработки документа | Останавливает обработку указанного документа | document\_id (path): ID документа |
| Повторный запуск обработки | Перезапускает обработку указанного документа | document\_id (path): ID документа |

Таблица 6 – Конечные точки раздела «Наборы данных»

| **Название** | **Описание** | **Параметры** |
| --- | --- | --- |
| Создание набора данных | Создает новый набор данных с фильтрами | filters (body): фильтры набора данных |
| Изменение набора данных | Обновляет существующий набор данных | dataset\_id (path): ID набора данных, filters (body): фильтры набора данных |
| Просмотр наборов данных | Возвращает список всех наборов данных | - |
| Анализ набора данных | Запускает анализ набора данных | dataset\_id (path): ID набора данных |
| Просмотр результатов анализа | Возвращает результаты анализа набора данных | dataset\_id (path): ID набора данных |
| Удаление набора данных | Удаляет указанный набор данных | dataset\_id (path): ID набора данных |

Таблица 7 – Конечные точки раздела «Модели»

| **Название** | **Описание** | **Параметры** |
| --- | --- | --- |
| Создание модели | Создает новую модель на основе набора данных | dataset\_id (body): ID набора данных |
| Просмотр моделей | Возвращает список всех моделей | - |

Продолжение таблицы 7

|  |  |  |
| --- | --- | --- |
| **Название** | **Описание** | **Параметры** |
| Анализ модели | Запускает анализ модели | model\_id (path): ID модели |
| Просмотр результатов анализа | Возвращает результаты анализа модели | model\_id (path): ID модели |
| Удаление модели | Удаляет указанную модель | model\_id (path): ID модели |
| Остановка построения модели | Останавливает процесс построения модели | model\_id (path): ID модели |
| Повторный запуск построения | Перезапускает процесс построения модели | model\_id (path): ID модели |

Таблица 8 – Конечные точки раздела «Прогнозы»

| **Название** | **Описание** | **Параметры** |
| --- | --- | --- |
| Создание прогноза | Создает новый прогноз на основе модели | model\_id (body): ID модели, dataset\_id (body): ID набора данных |
| Просмотр прогнозов | Возвращает список всех прогнозов | - |
| Анализ прогноза | Запускает анализ прогноза | forecast\_id (path): ID прогноза |
| Просмотр результатов анализа | Возвращает результаты анализа прогноза | forecast\_id (path): ID прогноза |
| Удаление прогноза | Удаляет указанный прогноз | forecast\_id (path): ID прогноза |
| Остановка прогнозирования | Останавливает процесс прогнозирования | forecast\_id (path): ID прогноза |
| Повторный запуск прогнозирования | Перезапускает процесс прогнозирования | forecast\_id (path): ID прогноза |

Таблица 9 – Конечные точки раздела «Настройки»

| **Название** | **Описание** | **Параметры** |
| --- | --- | --- |
| Просмотр настроек системы | Возвращает текущие настройки системы | - |
| Изменение настроек системы | Обновляет настройки системы | settings (body): параметры настроек |

При реализации серверной части был использован объектно-ориентированный подход для обеспечения бесшовной разработки и комплексного взаимодействия системы в целом. Диаграмма классов представлена на рисунке 20.

Таким образом, реализация этих конечных точек REST API обеспечит полный спектр функциональности, необходимой для взаимодействия пользователя с системой. Эти конечные точки позволят загружать и управлять патентными документами, создавать и анализировать наборы данных, строить модели и прогнозы, а также управлять настройками системы, обеспечивая удобный и интуитивно понятный интерфейс для пользователей.

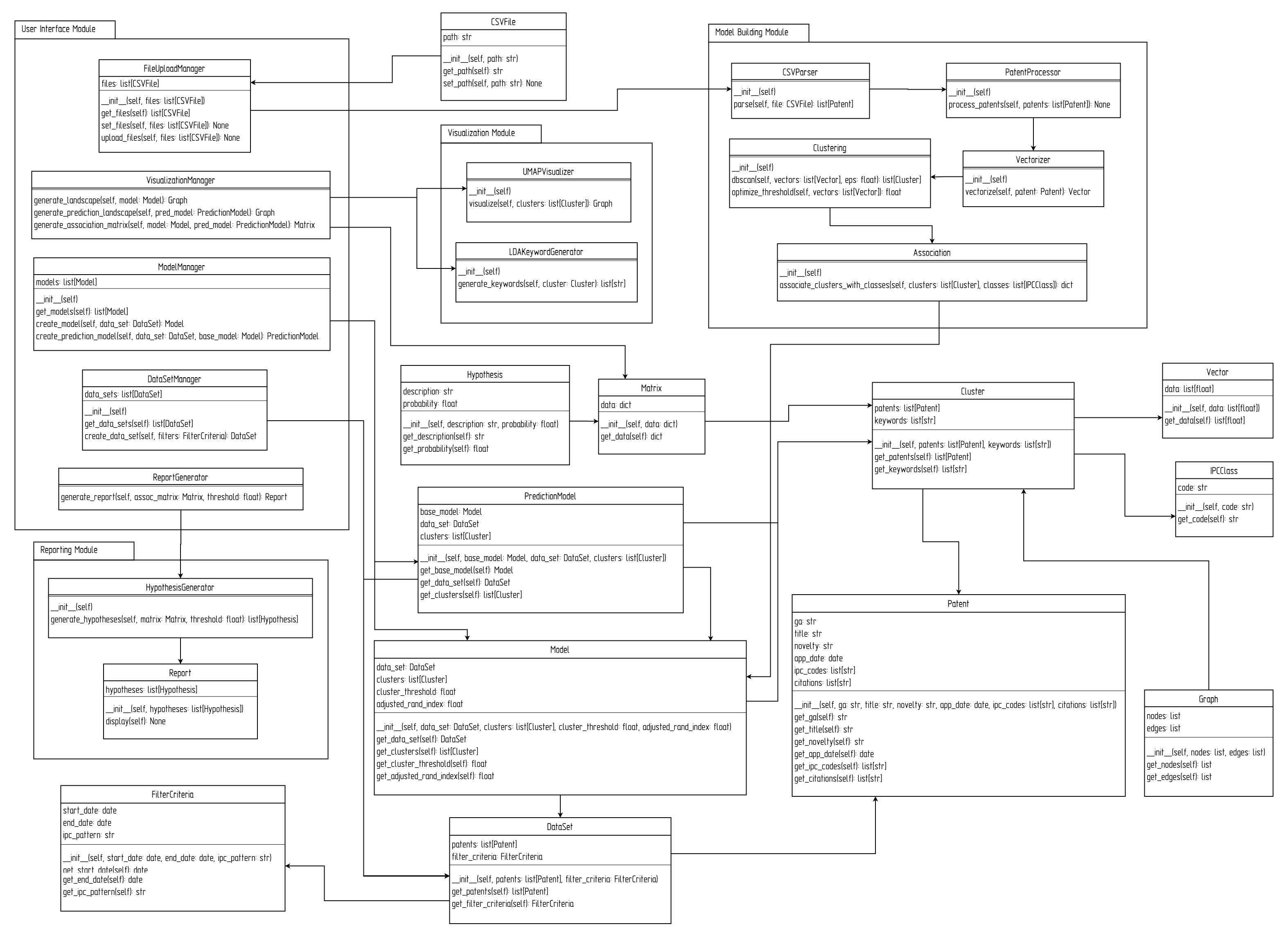


Рисунок 20 – Диаграмма классов программной системы

## 2.6. Разработка пользовательского интерфейса

Для разработки пользовательского интерфейса системы прогнозирования новых технологических областей был выбран фреймворк Vue.js с использованием языка TypeScript [69]. Этот выбор обусловлен удобством и высокой производительностью Vue.js, а также возможностью использования статической типизации TypeScript, что обеспечивает лучшую поддержку разработки и минимизацию ошибок.

Основные формы интерфейса включают форму загрузки данных, форму создания набора данных, форму анализа набора данных, форму создания модели, форму анализа модели, форму создания прогнозной модели, форму анализа прогнозной модели, форму управления настройками системы.

Форма загрузки данных позволяет пользователю загружать патентные документы в систему. Эта форма включает возможность выбора файлов для загрузки через стандартный файловый диалог или перетаскивание файлов в специально отведенную область. После выбора файлов пользователь может инициировать процесс загрузки, и система покажет статус загрузки и обработки каждого документа. В случае необходимости, пользователь может остановить обработку документа или повторно запустить ее.

Форма создания набора данных предоставляет пользователю возможность задать фильтры для формирования нового набора данных. Форма включает поля для указания диапазона дат, шаблонов классов IPC и других параметров фильтрации. После заполнения формы пользователь может сохранить набор данных, и система предоставит возможность просмотреть и проанализировать созданный набор.

Форма анализа набора данных позволяет пользователю запускать анализ выбранного набора данных. Пользователь выбирает нужный набор данных из списка доступных, после чего инициирует процесс анализа. Результаты анализа отображаются в виде различных графиков и диаграмм, которые можно скачать для дальнейшего использования.

Форма создания модели позволяет пользователю выбрать набор данных для построения модели. Пользователь выбирает набор данных из списка доступных и запускает процесс создания модели. Форма также предоставляет информацию о статусе построения модели и возможность остановки или повторного запуска процесса.

Форма анализа модели позволяет пользователю запускать и просматривать результаты анализа построенной модели. Пользователь выбирает модель из списка доступных, инициирует процесс анализа, и система отображает результаты в виде кластеров, графиков и ключевых слов.

Форма создания прогноза позволяет пользователю выбрать модель и набор данных для построения прогноза. Пользователь заполняет необходимые параметры, запускает процесс создания прогноза и может наблюдать за статусом выполнения. После завершения процесса пользователю доступны результаты анализа прогноза.

Форма анализа прогноза позволяет пользователю запускать и просматривать результаты анализа прогнозов. Пользователь выбирает прогноз из списка доступных, инициирует процесс анализа и получает результаты в виде диаграмм и графиков, отражающих прогнозируемые изменения и новые технологические области.

Форма управления настройками системы предоставляет пользователю возможность изменять и сохранять параметры системы. Пользователь может настроить различные аспекты системы, такие как параметры фильтрации данных, настройки визуализации и другие системные параметры. Изменения сохраняются и применяются немедленно.

## 2.7. Тестирование программного продукта

Тестирование программного продукта является важным этапом в разработке системы прогнозирования новых технологических областей на основе патентной информации. Оно необходимо для проверки правильности функционирования всех компонентов системы, их устойчивости к нагрузкам и общей производительности. Для достижения этих целей будут использоваться различные виды тестирования: функциональное и тестирование производительности:

1. функциональное тестирование – проверка того, что система функционирует в соответствии с заданными требованиями; включает тестирование всех функциональных компонентов системы, таких как загрузка документов, создание наборов данных, построение моделей и прогнозов;
2. тестирование производительности – измерение времени выполнения различных операций системы, таких как загрузка документов, обработка данных, создание моделей и прогнозов; оценивается скорость и эффективность работы системы.

### 2.7.1 Функциональное тестирование

Функциональное тестирование охватывает все основные функции системы, чтобы убедиться, что каждая из них работает корректно. Результаты функционального тестирования представлены в таблице 10.

Таблица 10 – Список результатов функционального тестирования

| **Функция** | **Действие** | **Результат** |
| --- | --- | --- |
| Загрузка документов | Загрузка правильного документа | Успешная загрузка, документ отображается в списке загруженных, начинается обработка |
| Загрузка документов | Загрузка неверного или поврежденного документа | Ошибка загрузки, отображение сообщения об ошибке |
| Загрузка документов | Просмотр статуса обработки документа | Документ проходит через состояния «ожидание», «в обработке», «завершено» |
| Создание набора данных | Создание набора данных с корректными параметрами | Успешное создание, новый набор данных отображается в списке доступных |
| Создание набора данных | Создание набора данных с некорректными параметрами | Ошибка создания, сообщение об ошибке |
| Создание набора данных | Изменение существующего набора данных | Успешное обновление, изменения сохраняются |
| Анализ набора данных | Запуск анализа набора данных | Успешный анализ, результаты отображаются корректно |
| Анализ набора данных | Запуск анализа набора данных без достаточного объема данных | Предупреждение или ошибка анализа |
| Создание модели | Создание модели на основе выбранного набора данных | Успешное создание, новая модель отображается в списке доступных |
| Создание модели | Создание модели с некорректными параметрами | Ошибка создания, сообщение об ошибке |
| Анализ модели | Запуск анализа модели | Успешный анализ, результаты отображаются корректно |
| Анализ модели | Запуск анализа модели с недостаточным объемом данных или ресурсов | Предупреждение или ошибка анализа |
| Создание прогноза | Создание прогноза на основе модели и набора данных | Успешное создание, новый прогноз отображается в списке доступных |
| Создание прогноза | Создание прогноза с некорректными параметрами | Ошибка создания, сообщение об ошибке |
| Управление настройками | Изменение настроек системы | Успешное сохранение и немедленное применение изменений |
| Управление настройками | Изменение настроек системы с некорректными параметрами | Ошибка сохранения, сообщение об ошибке |

### 2.7.2 Тестирование производительности

Тестирование производительности оценивает время выполнения различных операций системы. Результаты тестирования производительности представлены в таблице 11.

Таблица 11 – Список результатов тестирования производительности

| **Операция** | **Время выполнения** |
| --- | --- |
| Загрузка документа | В среднем 5 секунд |
| Полная обработка документа | В среднем 1 минута |
| Создание набора данных | В среднем 3 секунды |
| Анализ набора данных | В среднем 2 минуты |
| Создание модели | В среднем 5 минут |
| Анализ модели | В среднем 3 минуты |
| Создание прогноза | В среднем 4 минуты |
| Анализ прогноза | В среднем 2 минуты |

## Выводы

В результате выполнения работы по проектированию и реализации программного продукта достигнуты важные результаты в области автоматизированного анализа и прогнозирования новых технологических областей на основе патентной информации. Основные достижения включают разработку и интеграцию современных методов обработки естественного языка (NLP) и машинного обучения, что позволило создать точную и эффективную систему анализа патентных данных.

В рамках работы использованы модели SentenceTransformer для преобразования текстов в векторные представления, что улучшило качество анализа текстовых данных. Применение алгоритмов кластеризации, таких как DBSCAN, а также использование метрики Adjusted Rand Index для оптимизации порога кластеризации позволили эффективно группировать патенты по схожим характеристикам и выявлять новые технологические направления.

Также была разработана система визуализации патентного ландшафта с использованием методов UMAP и LDA, что дало возможность пользователям наглядно представлять структуру и взаимосвязи данных. Это способствует лучшему пониманию и интерпретации результатов анализа.

Процесс разработки включал определение архитектуры программной системы, описание основных компонентов и их взаимодействие. Созданы модули для загрузки и предобработки данных, преобразования текстовой информации в векторное представление, кластеризации данных и визуализации результатов. Программный продукт реализован с использованием языка программирования Python и современных библиотек для обработки данных и машинного обучения.

Функциональное тестирование системы показало высокую точность и эффективность в идентификации технологических трендов. Система позволяет загружать патентные данные в формате .csv, проводить их предварительную обработку, фильтрацию по заданным критериям и строить прогнозные модели для последующих периодов времени. Это облегчает процесс анализа и прогнозирования, повышая точность и обоснованность принимаемых решений.

# 3 Разработка методики тестирования и оценки качества прогнозирования системы

Система прогнозирования должна быть способна анализировать большие объемы патентных данных, выявлять скрытые закономерности и тенденции, а также предоставлять прогнозы о будущих технологических направлениях. Для достижения этой цели в систему вводятся документы с патентами, которые проходят через множество этапов обработки и анализа. Ожидается, что система будет способна автоматически обрабатывать входные данные и генерировать предсказания.

Таким образом, процесс тестирования и оценки системы прогнозирования включает в себя множество этапов, от подготовки данных до анализа результатов. Этот подход позволяет получить полное представление о возможностях системы и ее потенциале для применения в реальных условиях.

## 3.1 Разработка алгоритма тестирования

Методика тестирования системы прогнозирования появления новых технологий на основе патентной информации включает несколько этапов, которые позволяют последовательно оценить работу системы и получить достоверные результаты. На рисунке 21 представлен алгоритм тестирования, который описывает все шаги процесса от настройки системы до получения и оценки прогнозов.

Первый этап заключается в открытии интерфейса системы и выполнении настройки параметров системы. Эти настройки включают выбор параметров анализа, установку фильтров для отбора релевантных патентов и настройку моделей обработки данных. После выполнения настроек необходимо оценить корректность их сохранения. В случае обнаружения ошибок настройки корректируются и процесс повторяется до получения удовлетворительных результатов. Это обеспечивает, что система работает в соответствии с заданными параметрами и готова к обработке данных.

После успешной настройки системы загружается документ с патентными данными, который проходит обработку в системе. Этот этап важен для проверки возможности системы обрабатывать входные данные корректно и без ошибок. Важно оценить корректность обработки документа. Если документ обрабатывается успешно, процесс продолжается. В противном случае фиксируется проблема, и проводится анализ для последующего решения. Фиксация проблем и их устранение на ранних этапах помогает избежать ошибок в дальнейшем и гарантирует точность обработки данных.

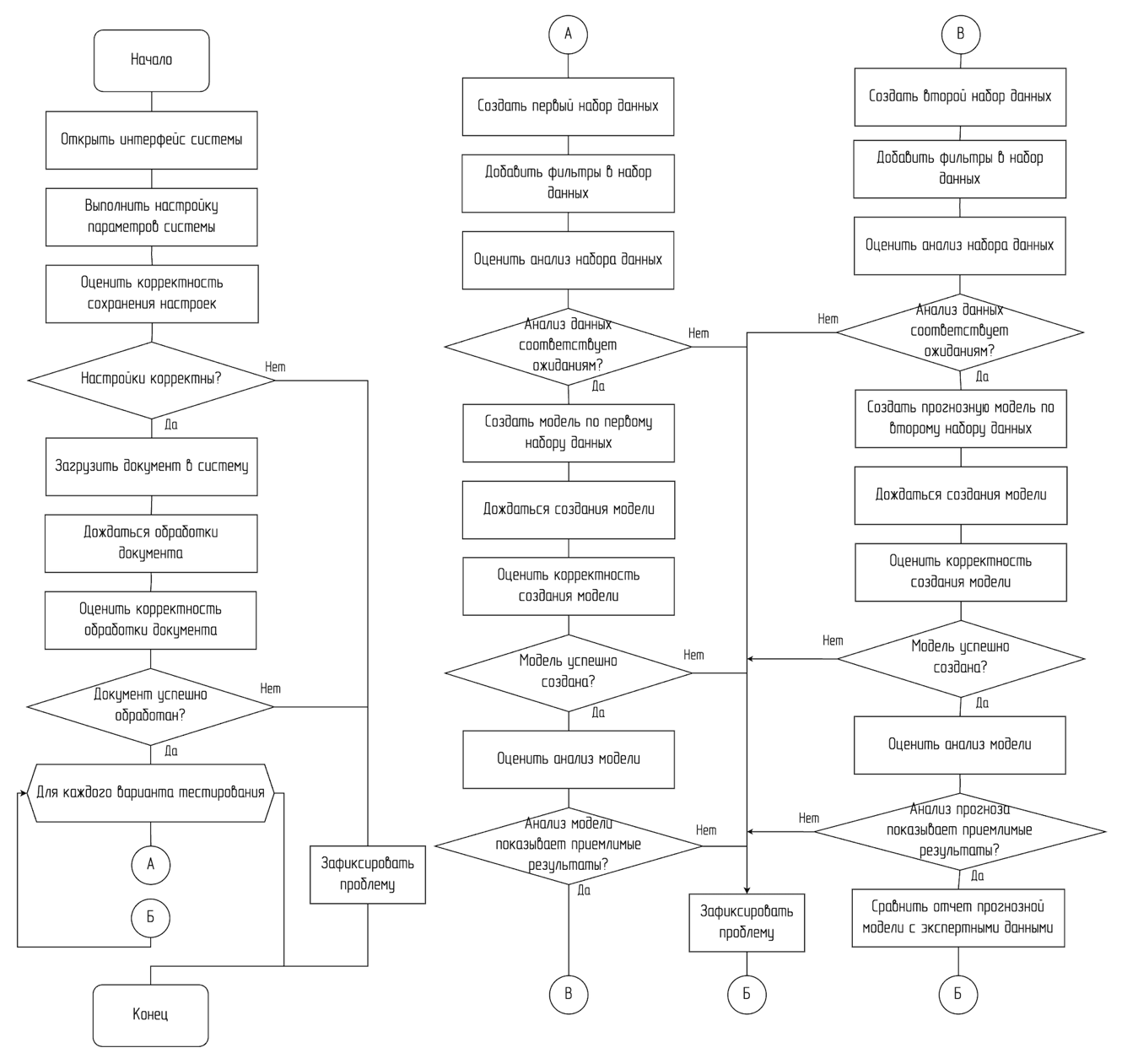


Рисунок 21 – Алгоритм тестирования оценки качества прогнозов

Следующий этап включает создание первого набора данных. Этот набор данных формируется на основе загруженных патентов и включает дополнительные фильтры для отбора данных, соответствующих заданным критериям. В набор данных добавляются необходимые фильтры, после чего проводится оценка его анализа. Если анализ данных соответствует ожиданиям, создается модель на основе первого набора данных. Этот процесс включает применение методов машинного обучения и аналитических моделей для выявления скрытых закономерностей и трендов в данных. После завершения создания модели проводится её оценка. Если результаты анализа модели являются приемлемыми, процесс продолжается, иначе фиксируется проблема и проводится доработка модели.

Для дальнейшего тестирования создается второй набор данных с аналогичными этапами фильтрации и оценки анализа. Этот набор данных может включать дополнительные параметры и данные для более детального анализа. После этого создается прогнозная модель на основе второго набора данных, которая также проходит оценку на соответствие ожидаемым результатам. Этот этап позволяет проверить, насколько точно система может прогнозировать будущие технологические тенденции на основе новых данных.

Заключительный этап тестирования включает сравнение отчета прогнозной модели с экспертными данными. Экспертные данные предоставляют независимую оценку точности и релевантности прогнозов, полученных системой. Это позволяет оценить, насколько предсказания системы соответствуют реальным трендам и ожиданиям в технологической сфере. Приемлемость результатов определяется на основе качественного анализа и экспертной оценки, что обеспечивает надежность и достоверность выводов.

Методика тестирования позволяет оценить корректность работы системы по следующим аспектам:

1. возможность настройки системы, что включает проверку правильности и сохранения установленных параметров;
2. корректность обработки документов, что проверяется на этапе загрузки и анализа патентных данных;
3. корректность построения моделей, что включает оценку результатов аналитических моделей и выявление трендов в данных;
4. корректность построения прогнозов, что оценивается путем сравнения предсказаний с реальными данными и экспертными оценками.

Кроме того, методика включает качественную оценку данных, полученных с помощью системы, которая охватывает следующие этапы:

1. анализ набора данных, что включает проверку полноты и релевантности отобранных данных;
2. анализ моделей, что включает оценку эффективности и точности аналитических моделей;
3. анализ прогнозов, что включает сравнение предсказаний с реальными данными и оценку их точности;
4. отчет о прогнозировании, что включает детальное описание полученных результатов и выводы по их анализу.

Эти этапы обеспечивают всестороннюю проверку и оценку системы, что позволяет выявить её сильные и слабые стороны, а также определить направления для дальнейшего улучшения. Такой подход гарантирует, что система будет надежным инструментом для прогнозирования технологических трендов и может быть использована для принятия стратегических решений в различных областях.

## 3.2 Набор данных

Для проведения тестирования и оценки системы прогнозирования появления новых технологий был использован набор данных, включающий патентные документы. Эти документы были собраны из открытых патентных баз данных, таких как Espacenet, Google Patents и других [70]. Основной задачей было собрать репрезентативный набор данных, который охватывал бы различные технологические области и временные периоды.

Набор данных включал патенты, опубликованные в период с 1995 по 2022 год. Каждый патент содержал информацию о тексте патента, дате подачи, международной патентной классификации (IPC), а также о дополнительных метаданных, таких как заявитель и изобретатель. На рисунке 22 показана структура патентного документа.

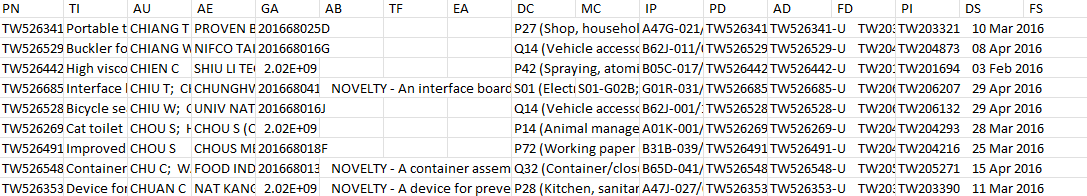


Рисунок 22 – Структура патентного документа

Для лучшего понимания структуры данных в таблице 12 приведен пример записи из патентного документа, который содержит основные поля, использующиеся в анализе.

Таблица 12 – Пример записи в патентном документе

|  |  |
| --- | --- |
| **Поле** | **Значение** |
| GA | 1994009953 |
| Title | Automatic system for teaching and monitoring |
| Novelty | The system comprises a programmable electronic... |
| AppDate | 1992-07-09 |
| Citations | 1996435891, 1994218095, 1996048511 |
| IPCs | G09B-007/06, G09B-005/02 |

Данный патентный документ включает несколько полей, которые были использованы в процессе анализа и прогнозирования. Эти поля позволяют идентифицировать патент и его содержание, а также классифицировать его по международной патентной классификации.

Набор данных состоит из следующих основных полей:

* код патента;
* название патента;
* новизна;
* дата подачи;
* международная патентная классификация (IPC);
* цитирования.

Набор данных включает более 100000 патентов, охватывающих широкий спектр технологических областей. Это позволило провести всесторонний анализ и оценку системы прогнозирования в различных контекстах. Данные охватывали такие области, как информационные технологии, медицинские устройства, химическая промышленность, энергетика и многие другие.

## 3.3 Метод оценки результатов тестирования

Для оценки результатов тестирования системы прогнозирования появления новых технологий на основе патентной информации был выбран метод экспертной оценки. Этот метод позволяет объективно оценить точность и релевантность предсказаний, сделанных системой, основываясь на знаниях и опыте специалистов в области патентов и технологий.

Метод экспертной оценки был выбран, так как прогнозирование новых технологий на основе патентной информации является многослойной задачей, включающей множество факторов, которые сложно формализовать и оценить количественно. Патентные данные разнообразны и включают различные аспекты, такие как технологические детали, рынок, инновационность и прочее. Автоматизированные методы оценки, основанные на математических моделях, не всегда способны учесть все эти нюансы. Экспертная оценка позволяет учитывать качественные аспекты прогнозирования, такие как инновационность технологий, рыночные тенденции и другие контекстные факторы. Эксперты могут проводить глубокий анализ и учитывать множество факторов, которые сложно формализовать в рамках автоматического анализа, что делает результаты более достоверными и значимыми. Также влияет отсутствие надежных автоматизированных методов, способных точно оценивать прогнозирование появления новых технологий на основе патентной информации.

Методика экспертной оценки позволяет объективно и точно оценить результаты работы системы прогнозирования появления новых технологий на основе патентной информации. Этот подход обеспечивает надежную и достоверную оценку, которая учитывает все значимые факторы и контекстные особенности, что делает результаты более полезными для практического применения. В дальнейшем результаты экспертной оценки могут быть использованы для улучшения и доработки системы, а также для разработки новых методов и подходов к анализу патентных данных.

## 3.4 Тестирование системы

Тестирование системы прогнозирования появления новых технологий на основе патентной информации включает несколько этапов. На каждом этапе используется интерфейс системы для выполнения необходимых операций. В данной главе будут описаны основные этапы тестирования: загрузка документа, создание наборов данных, построение моделей и построение прогнозов.

Для оценки системы были выбраны временные диапазоны от 1995 до 2022 года, и анализ проводился для различных классов Международной патентной классификации (IPC).

### 3.4.1 Загрузка документов

Первый этап тестирования включает загрузку патентного документа в систему. Для этого используется окно «Документы» (рисунок 23). В этом окне пользователю предоставляется возможность загрузить необходимые патентные документы для дальнейшего анализа.

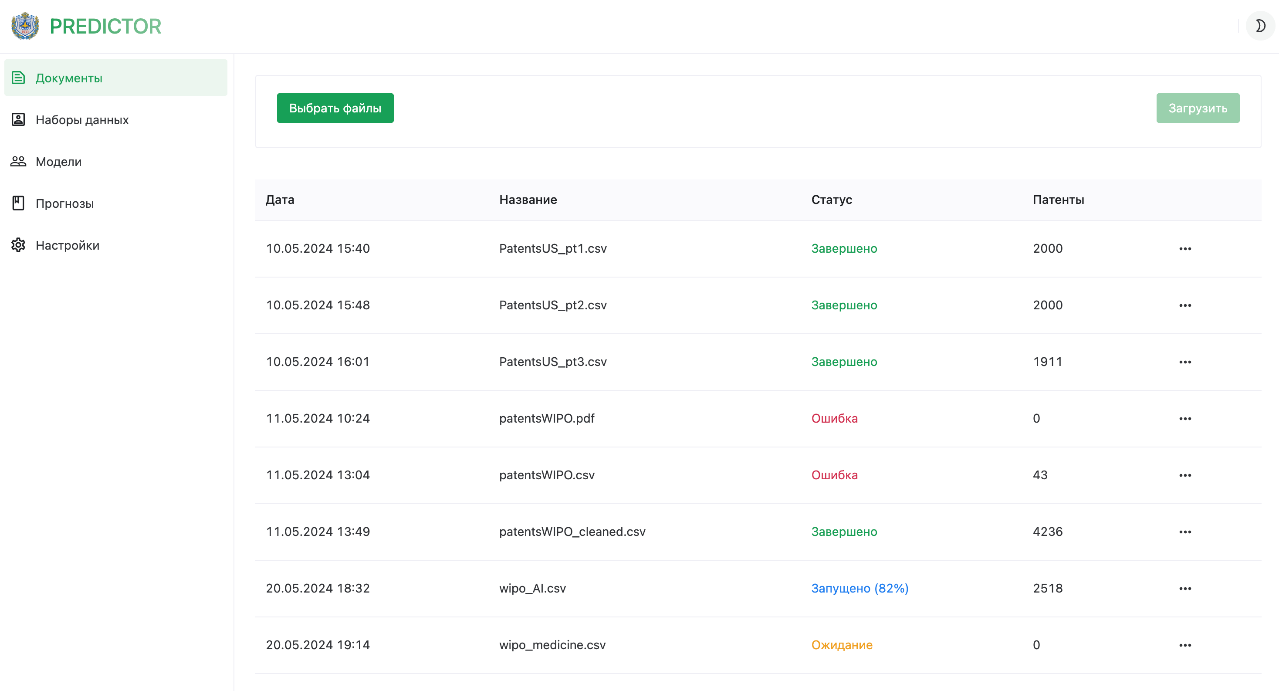


Рисунок 23 – Окно «Документы»

Загрузка документа включает следующие шаги:

* открытие окна «Документы»;
* выбор и загрузка патентного документа из локального хранилища;
* проверка корректности загрузки и отображения документа в системе.

### 3.4.2 Создание наборов данных

После загрузки патентных документов необходимо создать наборы данных, которые будут использоваться для анализа и построения моделей. Этот процесс выполняется в окне «Наборы данных» (рисунок 24). В этом окне пользователи могут создавать и управлять наборами данных, применяя различные фильтры и параметры.

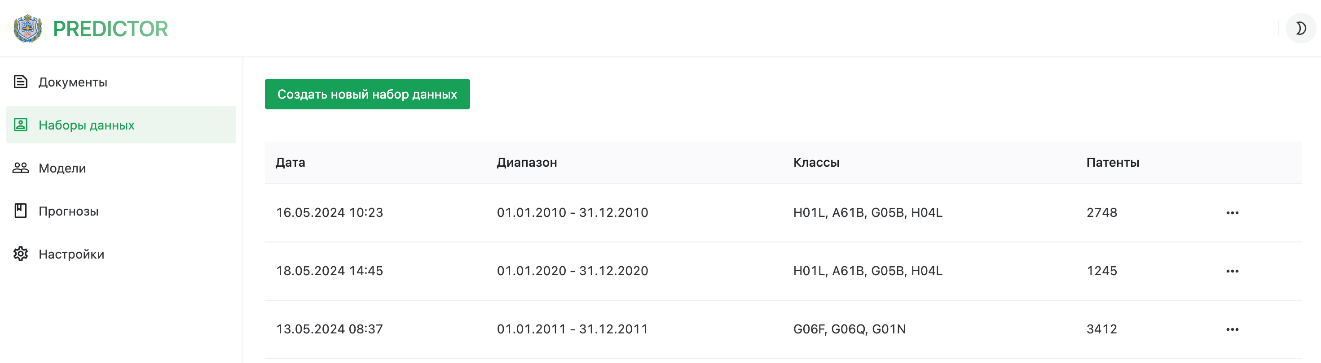


Рисунок 24 – Окно «Наборы данных»

Основные шаги включают:

* открытие окна «Наборы данных»;
* создание нового набора данных;
* применение фильтров и параметров для отбора релевантных данных (рисунок 25);
* просмотр анализа набора данных (рисунок 26).

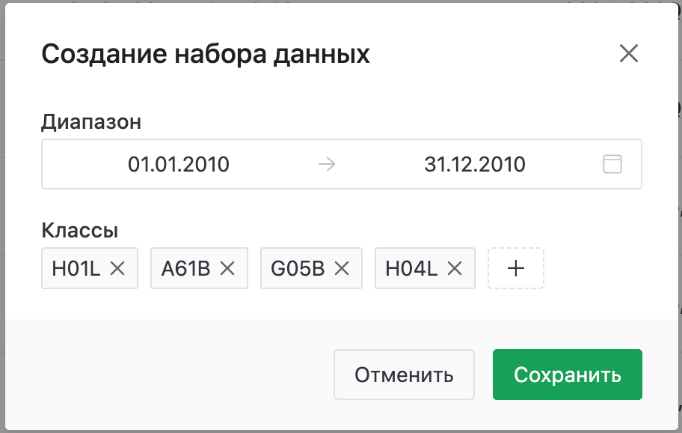


Рисунок 25 – Модальное окно создания набора данных

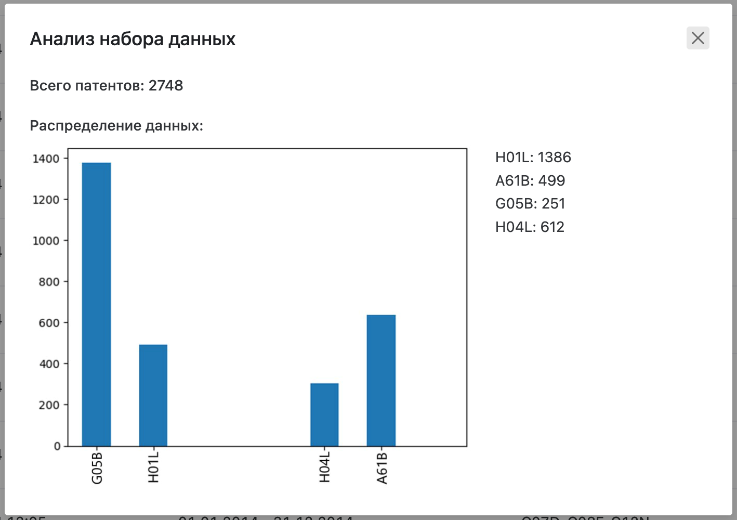


Рисунок 26 – Модальное окно анализа набора данных

### 3.4.4 Построение моделей

На следующем этапе выполняется построение моделей на основе созданных наборов данных. Для этого используется окно «Модели» (рисунок 27). В этом окне пользователи могут выбрать набор данных, на основе которого нужно построить модель патентного ландшафта.

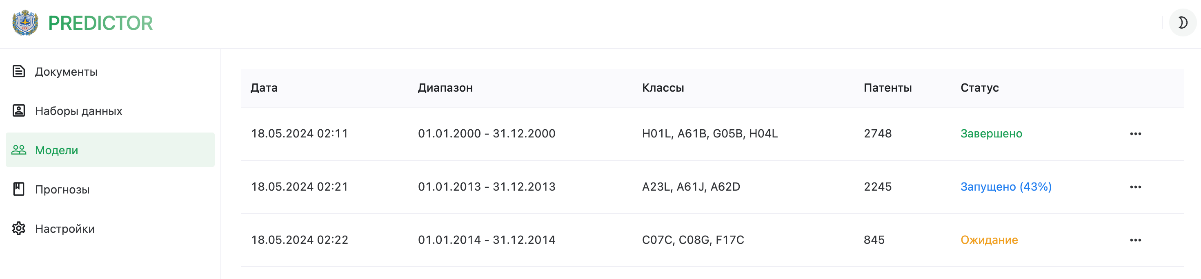


Рисунок 27 – Окно «Модели»

Основные шаги включают:

* открытие окна «Модели»;
* выбор метода моделирования и параметров модели;
* обучение модели на основе выбранного набора данных;
* анализ модели для оценки ее эффективности (рисунок 28).

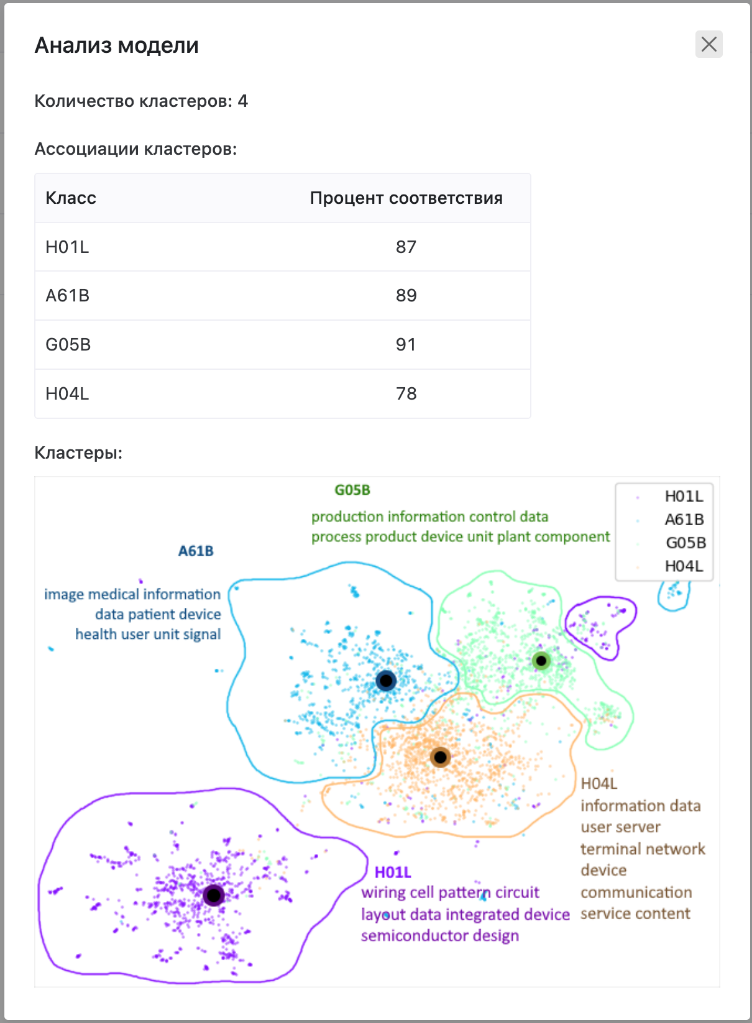


Рисунок 28 – Модальное окно анализа модели

### 3.4.4 Построение прогнозов

Заключительным этапом является построение прогнозов на основе обученных моделей. Этот процесс выполняется в окне «Прогнозы» (рисунок 29). Пользователи могут использовать обученные модели для прогнозирования появления новых технологий на основе созданных ранее моделей и наборов данных.

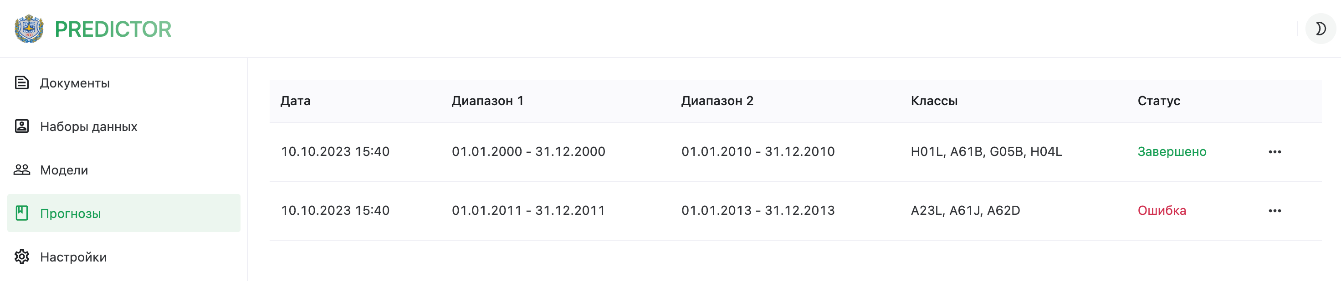


Рисунок 29 – Окно «Прогнозы»

Основные шаги включают:

* открытие окна «Прогнозы»;
* выбор обученной модели и параметров прогнозирования;
* получение прогнозов и их визуализация;
* анализ прогнозов для оценки их релевантности (рисунок 30).

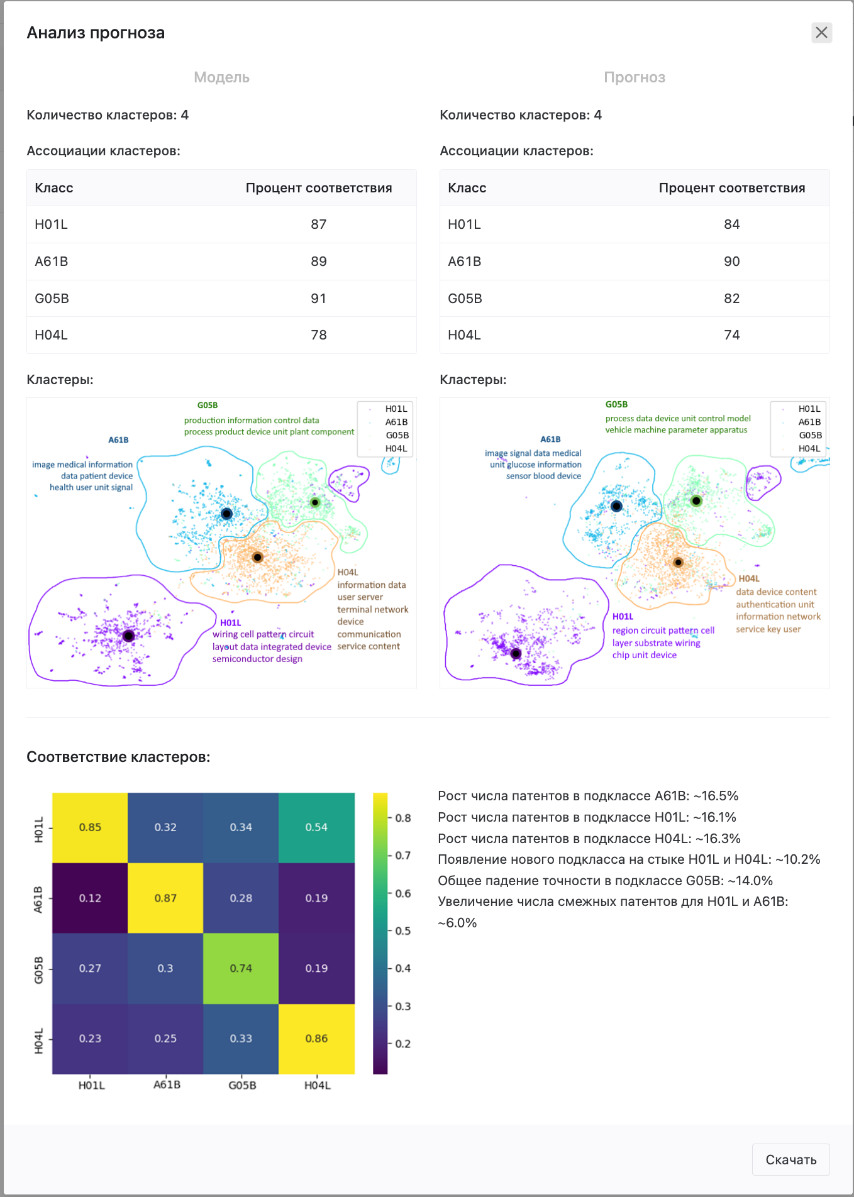


Рисунок 30 – Модальное окно анализа прогноза

## 3.5 Результаты тестирования

В процессе тестирования системы прогнозирования появления новых технологий на основе патентной информации были получены следующие результаты. Экспертная оценка была использована для верификации точности и релевантности предсказаний, сделанных системой.

Результаты тестирования представлены в таблице 13.

Таблица 13 – Результаты тестирования системы

| **№** | **Параметры** | **Описание** | **Результат** | **Соответствие** |
| --- | --- | --- | --- | --- |
| 1 | 2010 и 2020 года Подклассы 'H01L', 'A61B', 'G05B', 'H04L' | Большое количество патентов про обработку медицинских данных | Смещение фокуса в G05B в сторону A61B и H04L | Да |
| 2 | 2011 и 2021 года Подклассы 'G06F', 'G06Q', 'G01N' | Увеличение патентов в области вычислительных технологий и анализа данных | Постоянный рост в 'G06F' и 'G06Q' | Да |
| 3 | 2012 и 2020 года Подклассы 'H02J', 'H04B', 'H04W' | Резкий рост патентов в беспроводных и энергетических системах | Увеличение патентов в 'H04B' и 'H04W', снижение в 'H02J' | Нет |
| 4 | 2013 и 2019 года Подклассы 'B60L', 'B62D', 'F01N' | Рост патентов в области транспортных технологий | Значительное увеличение в 'B60L' и 'B62D' | Да |
| 5 | 2014 и 2020 года Подклассы 'C07D', 'C08F', 'C12N' | Рост патентов в области химических технологий и биотехнологий | Постоянный рост в 'C07D', 'C08F', 'C12N' | Да |
| 6 | 2015 и 2022 года Подклассы 'H03M', 'H04L', 'H05B' | Увеличение патентов в области коммуникационных технологий и освещения | Равномерное распределение между 'H03M' и 'H04L', снижение в 'H05B' | Нет |
| 7 | 2016 и 2021 года Подклассы 'A01B', 'A61K', 'B01D' | Резкий рост патентов в области сельского хозяйства и медицины | Увеличение патентов в 'A01B' и 'A61K', уменьшение в 'B01D' | Да |
| 8 | 2017 и 2020 года Подклассы 'E21B', 'F16H', 'G01V' | Увеличение патентов в области добычи полезных ископаемых | Постоянный рост в 'E21B', 'F16H', 'G01V' | Да |
| 9 | 2018 и 2022 года Подклассы 'H04W', 'H05K', 'G09G' | Рост патентов в области беспроводной связи и дисплейных технологий | Увеличение патентов в 'H04W', снижение в 'H05K', 'G09G' | Нет |
| 10 | 2011 и 2020 года Подклассы 'B65D', 'F16L', 'H01R' | Увеличение патентов в области упаковки и соединительных технологий | Значительный рост в 'B65D', стабильность в 'F16L', уменьшение в 'H01R' | Нет |

Продолжение таблицы 13

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Параметры** | **Описание** | **Результат** | **Соответствие** |
| 11 | 2010 и 2019 года Подклассы 'G06F', 'G06T', 'H04N' | Рост патентов в области информационных технологий и обработки изображений | Постоянный рост в 'G06F' и 'G06T', снижение в 'H04N' | Да |
| 12 | 2008 и 2018 года Подклассы 'B82Y', 'G02B', 'H01S' | Увеличение патентов в области нанотехнологий и оптики | Постоянный рост в 'B82Y' и 'G02B', стабильность в 'H01S' | Да |
| 13 | 2005 и 2015 года Подклассы 'C07C', 'C08G', 'F17C' | Рост патентов в области химической промышленности и хранения газа | Увеличение патентов в 'C07C', снижение в 'C08G' и 'F17C' | Нет |
| 14 | 2003 и 2013 года Подклассы 'A23L', 'A61J', 'A62D' | Увеличение патентов в области пищевой и медицинской промышленности | Постоянный рост в 'A23L' и 'A61J', снижение в 'A62D' | Да |
| 15 | 2004 и 2014 года Подклассы 'B29C', 'D06F', 'F42B' | Рост патентов в области обработки пластмасс и текстиля | Увеличение патентов в 'B29C', снижение в 'D06F' и 'F42B' | Нет |

Для оценки точности системы был использован показатель точности, рассчитываемый по следующей формуле:

В данном случае правильными предсказаниями считаются те, где результат соответствует ожиданиям, отмеченным в столбце «Соответствие».

Из 15 сценариев, 11 были отмечены как «Да» (правильные предсказания), и 4 как «Нет» (неправильные предсказания):

## Выводы

Целью данного этапа исследования было разработать методику, которая позволила бы объективно и всесторонне оценить эффективность предложенной системы.

Был разработан алгоритм тестирования и оценки. Этот алгоритм включает несколько последовательных шагов, которые позволяют оценить работу системы от загрузки данных до получения прогнозов и их последующей оценки.

Первый шаг алгоритма включает загрузку патентных документов в систему. Пользователь открывает интерфейс, выбирает необходимые документы и загружает их в систему. Система проверяет корректность загрузки и отображает документы для дальнейшего использования.

Второй шаг алгоритма посвящен созданию наборов данных. В этом этапе пользователь создает наборы данных из загруженных документов, применяя фильтры и параметры для отбора релевантных патентов. После создания наборы данных анализируются на полноту и соответствие заданным критериям.

На третьем шаге происходит построение моделей. Пользователь выбирает набор данных, после чего система обучает. Обученные модели анализируются на предмет их эффективности и точности.

Заключительный шаг включает построение прогнозов. С использованием обученных моделей пользователь выполняет прогнозирование новых патентов и технологических трендов. Прогнозы визуализируются и анализируются для оценки их точности и релевантности.

Результаты тестирования показали, что система прогнозирования способна корректно идентифицировать основные направления развития технологий в большинстве случаев. Экспертная оценка подтвердила высокую точность и релевантность предсказаний.

Таким образом, разработанная методика тестирования и оценки качества прогнозирования системы продемонстрировала свою эффективность и позволила всесторонне оценить работу системы. Это обеспечивает надежность и достоверность прогнозов, что делает систему полезным инструментом для анализа технологических трендов и стратегического планирования в различных областях науки и промышленности.

# ЗАКЛЮЧЕНИЕ

В результате выполнения выпускной квалификационной работы по созданию программной системы для анализа и прогнозирования новых технологических областей на основе патентной информации были успешно реализованы все поставленные задачи. Система позволяет эффективно анализировать большие массивы патентных данных и выявлять потенциальные направления технологического развития, что является критически важным для стратегического планирования и инновационной деятельности.

Исследовательская часть работы включала глубокий анализ существующих методов обработки текстовой информации. Были рассмотрены различные подходы и алгоритмы, используемые для анализа патентных данных, включая методы машинного обучения и обработки естественного языка (NLP). Особое внимание уделялось техникам векторизации текста, таким как TF-IDF, Word2Vec, и BERT, а также методам кластеризации данных, включая K-means, DBSCAN и иерархическую кластеризацию. Были исследованы возможности и ограничения каждого из методов, что позволило выбрать оптимальные подходы для разработки системы.

Конструкторская часть работы была направлена на разработку архитектуры программной системы. Были определены основные компоненты системы, их взаимодействие и функциональные возможности. Основу системы составляют модули для загрузки и предобработки данных, преобразования текстовой информации в векторное представление, кластеризации данных и визуализации результатов. Важной частью проектирования стало обеспечение масштабируемости и гибкости системы, что позволяет адаптировать ее под различные требования пользователей. В рамках этой части работы были разработаны алгоритмы для оптимизации порога кластеризации и формирования ассоциаций между патентами и технологическими направлениями. Также часть включала реализацию программной системы на языке Python с использованием современных библиотек для машинного обучения и анализа данных. Были разработаны и протестированы модули для обработки патентных данных, построения моделей и визуализации результатов. Система использует модель SentenceTransformer для преобразования текстов патентов в векторное представление, метод DBSCAN для кластеризации патентов и метрику Adjusted Rand Index для оптимизации порога кластеризации. Для визуализации патентного ландшафта применяются методы UMAP и LDA, что позволяет пользователям наглядно видеть структуру и взаимосвязи данных.

В технологической части была разработана методика тестирования точности результатов системы. Результаты тестирования показали высокую эффективность и точность работы системы. В ходе экспериментов система продемонстрировала способность корректно идентифицировать технологические тренды и формировать обоснованные гипотезы о появлении новых технологических областей. Система позволяет загружать патентные данные в формате .csv, проводить их предварительную обработку, фильтрацию по заданным критериям и строить прогнозные модели для последующих периодов времени. Пользователи могут создавать и визуализировать патентные ландшафты, анализировать изменения в данных и формулировать гипотезы о возможных направлениях технологического развития.

Проектная работа выполнена в соответствии с установленными сроками и полностью соответствует требованиям, описанным в техническом задании. При выполнении работы достигнуты все поставленные цели. Разработанная программная система готова к применению в реальных условиях и может существенно повысить эффективность процессов анализа и прогнозирования в области патентной информации.

Разработанная программная система предоставляет пользователям-экспертам необходимые инструменты для анализа, визуализации и прогнозирования, что способствует более информированному и обоснованному принятию стратегических решений в сфере технологического развития.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Создание интеллектуальных унифицированных систем поддержки принятия стратегических решений на базе глубокой и прогнозной аналитики больших данных. [Электронный ресурс]. // Отчет о НИР (промежуточный), МГТУ им. Н.Э. Баумана, руководитель Березкин Д.В., исполнители: Алфимцев А.Н. [и др.]. – М., 2024. – 308 с. – № ГР 224032800029-0. – Режим доступа: https://www.rosrid.ru/ikrbs/detail/BQOZPPS72GQ9AR3HFEAC0WKI (дата обращения: 23.05.2024).
2. Березкин Д.В., Козлов И.А., Мартынюк П.А., Панфилкин А.М., Киван Н. Система извлечения информации из текстовых документов с помощью нейронных сетей. // Искусственный интеллект в автоматизированных системах управления и обработки данных. Сборник статей Всероссийской научной конференции. В 2-х томах. Том 2. – Москва: МГТУ им. Н.Э. Баумана, 2022. – С. 254-260. – EDN NAHYPD.
3. Панфилкин А.М., Мартынюк П.А., Козлов И.А., Исянова А.М., Юнусов О.И. Обзор моделей представления данных для задач патентного анализа. // Искусственный интеллект в автоматизированных системах управления и обработки данных. Сборник статей II Всероссийской научной конференции. В 5 томах. – Москва: КДУ, Добросвет, 2023. – С. 310-315. – EDN TUXECT.
4. Hoshino Y., Utsumi Y., Matsuda Y., Tanaka Y., Nakata K. IPC prediction of patent documents using neural network with attention for hierarchical structure. [Электронный ресурс]. // PLoS One. – 2023. – Т. 18, № 3. – e0282361. – Режим доступа: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9980776/ (дата обращения: 27.05.2024). – DOI: 10.1371/journal.pone.0282361.
5. Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents. [Электронный ресурс]. // ResearchGate. – Режим доступа: https:// www.researchgate.net/publication/326425709\_Text\_Mining\_Use\_of\_TF-IDF\_to\_E xamine\_the\_Relevance\_of\_Words\_to\_Documents (дата обращения: 27.05.2024).
6. Word2Vec: What are Word Embedding? [Электронный ресурс]. // Digitate Blog. – Режим доступа: https://digitate.com/blog/word2vec-what-are-word-embedding/ (дата обращения: 27.05.2024).
7. Nguyen T., Ji S., Bai X. Automated Feature Engineering for Machine Learning: A Critical Survey. // Proceedings of the 14th ACM International Conference on Web Search and Data Mining (WSDM '21). – Association for Computing Machinery, New York, NY, USA, 2021. – P. 782–790. – DOI: 10.1145/3507971.3507972.
8. Word Embeddings using FastText. [Электронный ресурс]. // GeeksforGeeks. – Режим доступа: https://www.geeksforgeeks.org/word-embeddin gs-using-fasttext/ (дата обращения: 27.05.2024).
9. Reimers N., Gurevych I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. [Электронный ресурс]. // arXiv. – 2019. – Режим доступа: https://arxiv.org/pdf/1908.10084 (дата обращения: 27.05.2024).
10. Sarkar S., Feng D., Karmaker Santu S. Exploring Universal Sentence Encoders for Zero-shot Text Classification. [Электронный ресурс]. // Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 2: Short Papers). – Association for Computational Linguistics, 2022. – P. 135–147. – Режим доступа: https://aclanthology.org/2022. aacl-short.18/ (дата обращения: 27.05.2024).
11. What is the BERT language model? [Электронный ресурс]. // TechTarget. – Режим доступа: https://www.techtarget.com/searchenterpriseai/ definition/BERT-language-model (дата обращения: 27.05.2024).
12. Zong M., Krishnamachari B. A Survey on GPT-3. [Электронный ресурс]. // arXiv. – 2022. – Режим доступа: https://arxiv.org/pdf/2212.00857 (дата обращения: 27.05.2024).
13. K-Means Clustering in Machine Learning: A Deep Dive. [Электронный ресурс]. // DataScientest. – Режим доступа: https://datascientest.com/en/k-means-clustering-in-machine-learning-a-deep-dive (дата обращения: 27.05.2024).
14. Hierarchical Clustering. [Электронный ресурс]. // LearnDataSci. – Режим доступа: https://www.learndatasci.com/glossary/hierarchical-clustering (дата обращения: 27.05.2024).
15. Ester M., Kriegel H.-P., Sander J., Xu X. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. // Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD- 96). – 1996. – P. 226-231. – Режим доступа: https://www.dbs.ifi.lmu.de/ Publikationen/Papers/KDD-96.final.frame.pdf (дата обращения: 27.05.2024).
16. US Patent No. US-20150263925-A1. Method and System for Natural Language Processing. [Электронный ресурс]. // Unified Patents. – Режим доступа: https://portal.unifiedpatents.com/patents/patent/US-20150263925-A1 (дата обращения: 27.05.2024).
17. Anuranjana, Mittas N., Mehrotra D. Clustering the Patent Data Using K-Means Approach. [Электронный ресурс]. // In: Hoda M.N., Chauhan N., Quadri S.M.K., Srivastava P.R. (eds) Software Engineering. – Singapore: Springer, 2019. – P. 639-645. – Режим доступа: https://link.springer.com/chapter/10.1007/978-981-10-8848-3\_61 (дата обращения: 27.05.2024).
18. Jun S., Sung Park S., Sik Jang D. Technology forecasting using matrix map and patent clustering. [Электронный ресурс]. // Industrial Management & Data Systems. – 2012. – Т. 112, № 5. – С. 786-807. – Режим доступа: https:// www.emerald.com/insight/content/doi/10.1108/02635571211232352/full/html (дата обращения: 27.05.2024).
19. Hu J., Li S., Hu J., Yang G. A Hierarchical Feature Extraction Model for Multi-Label Mechanical Patent Classification. [Электронный ресурс]. // University of South Carolina, Scholar Commons. – 2018. – Режим доступа: https://schol arcommons.sc.edu/cgi/viewcontent.cgi?article=1263&context=csce\_facpub (дата обращения: 27.05.2024).
20. Frey B.J., Dueck D. Clustering by Passing Messages Between Data Points. [Электронный ресурс]. // Science. – 2007. – Т. 315, № 5814. – С. 972- 976. – Режим доступа: https://www.science.org/doi/abs/10.1126/science.1136800 (дата обращения: 27.05.2024).
21. Blei D.M., Ng A.Y., Jordan M.I. Latent Dirichlet Allocation. [Электронный ресурс]. // Journal of Machine Learning Research. – 2003. – Т. 3. – С. 993-1022. – Режим доступа: https://www.jmlr.org/papers/volume3/blei03a/ blei03a.pdf (дата обращения: 27.05.2024).
22. Wang J., Hsu C.-C. A topic-based patent analytics approach for exploring technological trends in smart manufacturing. [Электронный ресурс]. // Journal of Manufacturing Technology Management. – 2021. – Т. 32, № 1. – С. 110-135. – Режим доступа: https://www.emerald.com/insight/content/doi/10.1108/JMTM-03-2020-0106/full/html?skipTracking=true (дата обращения: 27.05.2024).
23. Slimani T., Lazzez A. Efficient Analysis of Pattern and Association Rule Mining Approaches. [Электронный ресурс]. // International Journal of Information Technology and Computer Science (IJITCS). – 2014. – Т. 6, № 3. – С. 70-81. – Режим доступа: https://arxiv.org/pdf/1402.2892 (дата обращения: 27.05.2024).
24. Jurafsky D., Martin J.H. Hidden Markov Models. [Электронный ресурс]. // Speech and Language Processing. – 2023. – Режим доступа: https://web. stanford.edu/~jurafsky/slp3/A.pdf (дата обращения: 27.05.2024).
25. Lee K., Go D., Park I., Yoon B. Exploring Suitable Technology for Small and Medium-Sized Enterprises (SMEs) Based on a Hidden Markov Model Using Patent Information and Value Chain Analysis. [Электронный ресурс]. // Sustainability. – 2017. – Т. 9, № 7. – 1100. – Режим доступа: https://www.mdpi. com/2071-1050/9/7/1100 (дата обращения: 27.05.2024).
26. Aristodemou L., Tietze F. The state-of-the-art on Intellectual Property Analytics (IPA): A literature review on artificial intelligence, machine learning and deep learning methods for analysing intellectual property (IP) data. [Электронный ресурс]. // World Patent Information. – 2018. – Т. 55. – С. 37-51. – Режим доступа: https://www.sciencedirect.com/science/article/pii/S0172219018300103 (дата обращения: 27.05.2024). – DOI: 10.1016/j.wpi.2018.07.002.
27. Vassiliou P.-C.G., Georgiou A.C. Markov and Semi-Markov Chains, Processes, Systems, and Emerging Related Fields. [Электронный ресурс]. // Mathematics. – 2021. – Т. 9, № 19. – 2490. – Режим доступа: https://www.mdpi. com/2227-7390/9/19/2490 (дата обращения: 27.05.2024).
28. Gradient-Boosted Decision Trees (GBDT). [Электронный ресурс]. // C3 AI. – Режим доступа: https://c3.ai/glossary/data-science/gradient-boosted-decision-trees-gbdt/ (дата обращения: 27.05.2024).
29. Li Y., Wang X., Chen C., Jing C., Wu T. Exploring firms’ innovation capabilities through learning systems. [Электронный ресурс]. // Neurocomputing. – 2020. – Т. 409. – С. 27-34. – Режим доступа: https://www.sciencedirect.com/ science/article/pii/S0925231220305312 (дата обращения: 27.05.2024). – DOI: 10.1016/j.neucom.2020.03.100.
30. What Is Random Forest? [Электронный ресурс]. // IBM. – Режим доступа: https://www.ibm.com/topics/random-forest (дата обращения: 27.05.2024).
31. European Patent Application. EP 3 826 029 A1. Workflow Predictive Analytics Engine. GE Precision Healthcare LLC, Wauwatosa, WI, US. Date of publication: 26.05.2021. [Электронный ресурс]. – Режим доступа: https://data. epo.org/publication-server/rest/v1.0/publication-dates/20210526/patents/EP3826029 NWA1/document.pdf (дата обращения: 27.05.2024).
32. Alzubaidi L., Zhang J., Humaidi A.J., Al-Dujaili A., Duan Y., Al-Shamma O., Santamaría J., Fadhel M.A., Al-Amidie M., Farhan L. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. [Электронный ресурс]. // Journal of Big Data. – 2021. – Т. 8, № 1. – 53. – Режим доступа: https://journalofbigdata.springeropen.com/articles/10.1186/s40537-021-00444-8 (дата обращения: 27.05.2024).
33. Popescu M.-C., Balas V., Perescu-Popescu L., Mastorakis N. Multilayer perceptron and neural networks. [Электронный ресурс]. // WSEAS Transactions on Circuits and Systems. – 2009. – Т. 8. – Режим доступа: https://www.researchgate. net/publication/228340819\_Multilayer\_perceptron\_and\_neural\_networks (дата обращения: 27.05.2024).
34. Activation Functions in Neural Networks [12 Types & Use Cases]. [Электронный ресурс]. // V7 Labs. – 2021. – Режим доступа: https://www.v7labs. com/blog/neural-networks-activation-functions (дата обращения: 27.05.2024).
35. Lee C., Kwon O., Kim M., Kwon D. Early identification of emerging technologies: A machine learning approach using multiple patent indicators. [Электронный ресурс]. // Technological Forecasting and Social Change. – 2017. – Т. 127. – DOI: 10.1016/j.techfore.2017.10.002. – Режим доступа: https://www. researchgate.net/publication/321192243\_Early\_identification\_of\_emerging\_technologies\_A\_machine\_learning\_approach\_using\_multiple\_patent\_indicators (дата обращения: 27.05.2024).
36. Purwono, Ma'arif A., Rahmaniar W., Imam H., Fathurrahman H.I.K., Frisky A., Haq Q.M.U. Understanding of Convolutional Neural Network (CNN): A Review. [Электронный ресурс]. // International Journal of Research in Computer Science. – 2023. – Т. 2. – С. 739-748. – DOI: 10.31763/ijrcs.v2i4.888. – Режим доступа: https://www.researchgate.net/publication/367157330\_Understanding\_of\_ Convolutional\_Neural\_Network\_CNN\_A\_Review (дата обращения: 27.05.2024).
37. Ji T., Self N., Fu K., Chen Z., Ramakrishnan N., Lu C.-T. Citation Forecasting with Multi-Context Attention-Aided Dependency Modeling. [Электронный ресурс]. // Association for Computing Machinery. – 2024. – Т. 18, № 6. – DOI: 10.1145/3649140. – Режим доступа: https://doi.org/10.1145/3649140 (дата обращения: 27.05.2024).
38. Das S., Tariq A., Santos T., Kantareddy S.S., Banerjee I. Recurrent Neural Networks (RNNs): Architectures, Training Tricks, and Introduction to Influential Research. [Электронный ресурс]. // В кн.: Colliot O. (ред.) Machine Learning for Brain Disorders. Neuromethods, том 197. – New York, NY: Humana, 2023. – DOI: 10.1007/978-1-0716-3195-9\_4. – Режим доступа: https://link.springer.com/ protocol/10.1007/978-1-0716-3195-9\_4 (дата обращения: 27.05.2024).
39. Ji T., Self N., Fu K., Lu C.-T. Citation Predicting with Multi-Context Attention-Aided Dependency Modeling. [Электронный ресурс]. // Association for Computing Machinery. – 2024. – Т. 18, № 6. – DOI: 10.1145/3649140. – Режим доступа: https://doi.org/10.1145/3649140 (дата обращения: 27.05.2024).
40. Staudemeyer R.C., Morris E.R. Understanding LSTM – A Tutorial into Long Short-Term Memory Recurrent Neural Networks. [Электронный ресурс]. // arXiv. – 2019. – Режим доступа: https://arxiv.org/abs/1909.09586 (дата обращения: 27.05.2024).
41. Hou J., Tang S., Zhang Y. A novel technology life cycle analysis method based on LSTM and CRF. [Электронный ресурс]. // Scientometrics. – 2024. – Т. 129. – С. 1173-1196. – DOI: 10.1007/s11192-024-04946-z. – Режим доступа: https://link.springer.com/article/10.1007/s11192-024-04946-z (дата обращения: 27.05.2024).
42. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser L., Polosukhin I. Attention Is All You Need. [Электронный ресурс]. // arXiv. – 2017. – Режим доступа: https://arxiv.org/abs/1706.03762 (дата обращения: 27.05.2024).
43. Zhao J., Dong Z., Yao X., et al. Optimizing collaboration decisions in technological innovation through machine learning: identify trend and partners in collaboration-knowledge interdependent networks. [Электронный ресурс]. // Annals of Operations Research. – 2024. – DOI: 10.1007/s10479-024-05867-z. – Режим доступа: https://link.springer.com/article/10.1007/s10479-024-05867-z (дата обращения: 27.05.2024).
44. Jun S., Park S.S., Jang D.S. Patent Management for Technology Forecasting: A Case Study of the Bio-Industry. // Journal of Intellectual Property Rights. – 2012. – Т. 17, № 6. – С.
45. Sunghae J. A new patent analysis using association rule mining and boxjenkins modeling for technology forecasting. [Электронный ресурс]. // Information (Japan). – 2013. – Т. 16. – С. 555-562. – Режим доступа: https:// www.researchgate.net/publication/289640149\_A\_new\_patent\_analysis\_using\_association\_rule\_mining\_and\_boxjenkins\_modeling\_for\_technology\_forecasting (дата обращения: 27.05.2024).
46. Jeong Y., Jang H., Yoon B. Developing a risk-adaptive technology roadmap using a Bayesian network and topic modeling under deep uncertainty. [Электронный ресурс]. // Scientometrics. – 2021. – Т. 126. – С. 3697–3722. – DOI: 10.1007/s11192-021-03945-8. – Режим доступа: https://link.springer.com/ article/10.1007/s11192-021-03945-8 (дата обращения: 27.05.2024).
47. Березкин Д.В., Козлов И.А., Мартынюк П.А. Гибридный подход к прогнозированию возникновения и развития новых направлений в научно-технической сфере. // Гибридные и синергетические интеллектуальные системы: Материалы VI Всероссийской Поспеловской конференции с международным участием, Калининград, 27 июня – 01 2022 года. – Калининград: Балтийский федеральный университет имени Иммануила Канта, 2022. – С. 143-150. – DOI: 10.5922/978-5-9971-0687-4-5. – EDN EFKJME.
48. Berezkin D.V., Kozlov I.A., Martynyuk P.A., Panfilkin A.M. A method for creating structural models of text documents using neural networks. // Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. – 2023. – Т. 12, № 1. – С. 28-45. – DOI: 10.14529/cmse230102. – EDN TNLVQY.
49. Al-Debagy O., Martinek P. A Comparative Review of Microservices and Monolithic Architectures. [Электронный ресурс]. // arXiv. – 2019. – Режим доступа: https://arxiv.org/abs/1905.07997 (дата обращения: 27.05.2024).
50. Monolithic vs. SOA vs. Microservices vs. Serverless Architecture. [Электронный ресурс]. // RubyGarage. – Режим доступа: https://rubygarage.org/ blog/monolith-soa-microservices-serverless (дата обращения: 27.05.2024).
51. Erickson J., Siau K. Service Oriented Architecture: A Research Review from the Software and Applications Perspective. [Электронный ресурс]. // Innovations in Information Systems Modeling: Methods and Best Practices. – 2009. – С. 190-203. – ISBN: 9781605662794. – DOI: 10.4018/978-1-60566- 278-7.ch010. – Режим доступа: https://www.researchgate.net/publication/28820 4978\_Service\_Oriented\_Architecture\_A\_Research\_Review\_from\_the\_Software\_and\_Applications\_Perspective (дата обращения: 27.05.2024).
52. Liu G., Huang B., Liang Z., Qin M., Zhou H., Li Z. Microservices: architecture, container, and challenges. [Электронный ресурс]. // В кн.: 2020 IEEE 20th International Conference on Software Quality, Reliability and Security Companion (QRS-C). – 2020. – С. 629-635. – DOI: 10.1109/QRS-C51114.2020. 00107. – Режим доступа: https://ieeexplore.ieee.org/document/9282637 (дата обращения: 27.05.2024).
53. Python Programming Language. [Электронный ресурс]. // Python Software Foundation. – Режим доступа: https://www.python.org/ (дата обращения: 27.05.2024).
54. JavaScript Guide. [Электронный ресурс]. // Mozilla Developer Network. – Режим доступа: https://developer.mozilla.org/en-US/docs/Web/ JavaScript (дата обращения: 27.05.2024).
55. Schuster D. React vs. Vue: Comparing JavaScript Frameworks from a Beginner’s Perspective. [Электронный ресурс]. // KTH Royal Institute of Technology. – Degree Project in Computer Engineering. – Stockholm, Sweden, 2023. – 15 credits. – Режим доступа: https://kth.diva-portal.org/smash/get/diva2: 1789089/FULLTEXT01.pdf (дата обращения: 27.05.2024).
56. SentenceTransformers Documentation. [Электронный ресурс]. // SentenceTransformers. – Режим доступа: https://www.sbert.net/ (дата обращения: 27.05.2024).
57. Scikit-learn: Machine Learning in Python. [Электронный ресурс]. // Scikit-learn. – Режим доступа: https://scikit-learn.org/stable/ (дата обращения: 27.05.2024).
58. Uniform Manifold Approximation and Projection (UMAP). [Электронный ресурс]. // UMAP Documentation. – Режим доступа: https://umap-learn.readthedocs.io/en/latest/ (дата обращения: 27.05.2024).
59. Gensim: Topic modeling for humans. [Электронный ресурс]. // Gensim. – Режим доступа: https://github.com/piskvorky/gensim (дата обращения: 27.05.2024).
60. Using Plotly in Python. [Электронный ресурс]. // Geo-Python. – Режим доступа: https://geo-python.github.io/2017/lessons/L7/using-plotly.html (дата обращения: 27.05.2024).
61. MinIO: High Performance, Kubernetes Native Object Store. [Электронный ресурс]. // MinIO. – Режим доступа: https://min.io/ (дата обращения: 27.05.2024).
62. PostgreSQL: The World's Most Advanced Open-Source Relational Database. [Электронный ресурс]. // PostgreSQL. – Режим доступа: https://www. postgresql.org/ (дата обращения: 27.05.2024).
63. Fielding R.T. Representational State Transfer (REST). [Электронный ресурс]. // Ph.D. dissertation, University of California, Irvine. – 2000. – Режим доступа: https://ics.uci.edu/~fielding/pubs/dissertation/rest\_arch\_style.htm (дата обращения: 27.05.2024).
64. FastAPI Documentation. [Электронный ресурс]. // FastAPI. – Режим доступа: https://fastapi.tiangolo.com/ (дата обращения: 27.05.2024).
65. Liu Q., Sun X. Research of Web Real-Time Communication Based on Web Socket. [Электронный ресурс]. // International Journal of Communications, Network and System Sciences. – 2012. – Т. 05. – С. 797-801. – DOI: 10.4236/ ijcns.2012.512083. – Режим доступа: https://www.researchgate.net/publication/27 6491172\_Research\_of\_Web\_Real-Time\_Communication\_Based\_on\_Web\_Socket (дата обращения: 27.05.2024).
66. Centrifugo: Scalable real-time messaging server. [Электронный ресурс]. // Centrifugal Labs. – Режим доступа: https://centrifugal.dev/ (дата обращения: 27.05.2024).
67. OpenAPI Initiative. [Электронный ресурс]. // OpenAPI. – Режим доступа: https://www.openapis.org/ (дата обращения: 27.05.2024).
68. Docker: Enterprise Container Platform. [Электронный ресурс]. // Docker. – Режим доступа: https://www.docker.com/ (дата обращения: 27.05.2024).
69. TypeScript Documentation. [Электронный ресурс]. // TypeScript. – Режим доступа: https://www.typescriptlang.org/ (дата обращения: 27.05.2024).
70. USPTO Bulk Data Storage System. [Электронный ресурс]. // United States Patent and Trademark Office. – Режим доступа: https://bulkdata.uspto.gov/ (дата обращения: 27.05.2024).

# ПРИЛОЖЕНИЕ А

Техническое задание

Листов 9

# ПРИЛОЖЕНИЕ Б

Исходный текст программного модуля

Листов 21

# ПРИЛОЖЕНИЕ В

Графические материалы

Листов 10 формата А1

* схема структурная информационной системы;
* диаграмма вариантов использования;
* результаты сравнительного анализа подходов к прогнозированию;
* общая схема предлагаемого подхода к прогнозированию;
* концептуальная модель предметной области;
* схема алгоритма прогнозирования;
* схемы алгоритмов основных модулей системы;
* диаграмма классов предметной области;
* алгоритм тестирования модели прогнозирования;
* результаты тестирования системы.

|  |  |  |  |
| --- | --- | --- | --- |
| Копии верны |  |  | Д.В. Берёзкин |
|  | (подпись, дата) |  | (И.О. Фамилия) |