**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
ДОНЕЦКОЙ НАРОДНОЙ РЕСПУБЛИКИ
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

На правах рукописи

УДК 004. 89

Бурлаева Екатерина Игоревна

СОВЕРШЕНСТВОВАНИЕ МЕТОДОВ СИСТЕМНОГО АНАЛИЗА В ЗАДАЧАХ УПРАВЛЕНИЯ И СИСТЕМАТИЗАЦИИ СПЕЦИАЛИЗИРОВАННОЙ ИНФОРМАЦИИ

05.13.01– Системный анализ, управление и обработка информации
(по отраслям)

Диссертация
на соискание учёной степени кандидата
технических наук

Научный руководитель
доктор технических наук,
доцент Зори С. А.

Идентичность всех экземпляров
ПОДТВЕРЖДАЮ
Ученый секретарь диссертационного
совета Д 01.024.04
канд. техн. наук Т.В. Завадская

Донецк – 2019

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Актуальность темы исследования. Обмен информацией является важнейшей частью современной деятельности человека. По оценкам экспертов около 70% накопленной и используемой информации находятся в несистематизированной текстовой форме, вследствие чего затрудняется получение требуемых сведений по конкретной тематике. Таким образом, возникает острая необходимость в создании систем, позволяющих автоматически систематизировать специализированную информацию [1 с. 3].

В этой ситуации особую актуальность приобретают работы по созданию систем анализа, систематизации и управления специализированной текстовой информацией, так как даже высококвалифицированные эксперты испытывают затруднения по организации поиска документов и распределении полученных текстовых данных по тематикам [2].

Одним из способов систематизации, управления данными и их анализа является классификация информации, состоящая из сортировки текстовых документов по заранее определенным категориям [3 с. 101].

Связь работы с научными программами, планами, темами. В основу диссертационного исследования положены работы, выполненные в Донецком Национальном техническом университете в рамках научно-исследовательской работы кафедры искусственного интеллекта и системного анализа Г/Т № Н 17-18 «Информационные технологии в системах моделирования и управления организационными и техническими объектами»; Г/Т № Н 17-13 «Разработка теоретических способов и методов создания современных информационных систем», в которых соискатель принимал участие как исполнитель.

Степень разработанности темы исследования. В исследованиях, посвященных применению методов машинного обучения для классификации текстов, применяются в основном универсальные алгоритмы, которые применимы для широкого круга задач анализа, управления и обработки

специализированной информации [4 с. 213]. Качество рубрикации для систем, основанных на машинном обучении, является довольно высоким для небольших рубрикаторов, и значительно уменьшается с увеличением количества рубрик и усложнением структуры классификатора [5 с. 25].

Цель методов машинного обучения для задачи классификации текстовых документов заключается в построении модели классификации на основе обучающего набора и применении ее для предсказания класса или набора классов, релевантных для нового документа. Разработке и тестированию моделей данного вида, а также связанными с ней алгоритмами обработки текстовой информации в настоящее время посвящены труды таких авторов как Агеев М.С., Кураленок И.Е., Joachims T., Schapire R.E., Schutze H., Scbastiani F и др. [6, 7, 8, 9].

Исследования показали, что эффективная обработка и анализ специализированной информации практически невозможны без разработки комплексной модели процесса систематизации на основе машинного обучения и знаний эксперта, т.е. парадигматических и синтагматических подходов, т.к. состав и содержимое анализируемых документов постоянно изменяется.

В условиях роста информационного пространства и необходимости автоматизации информационных процессов повышение эффективности существующих моделей и методов для построения систем классификации и управления текстовой информацией является важной и актуальной научно–технической задачей, имеющей отраслевое значение.

Целью исследования является модернизация моделей, методик и алгоритмов системного анализа, искусственного интеллекта и компьютерной лингвистики для повышения эффективности автоматизации, систематизации специализированной текстовой информации больших объёмов с учетом особенностей русскоязычных текстовых документов.

Для достижения цели диссертационного исследования поставлены и решены следующие **задачи**:

1. Проанализировать существующие методы, модели и алгоритмы, используемые для решения задач систематизации и управления специализированной текстовой информации.

2. Разработать модифицированную модель автоматической обработки специализированной информации, основываясь на объединении достоинств парадигматических и синтагматических подходов.

3. Выявить закономерности изменения качества работы модели обработки текстовой информации при внесении изменений в ее конфигурацию.

4. Осуществить исследование выполненной разработки усовершенствованной модели обработки текстовой информации.

5. Обосновать подход к повышению эффективности применения предложенных моделей и методов для систем автоматической обработки специализированной информации.

6. Проанализировать качество работы предложенных средств при решении практических задач классификации текстов.

Объект исследования – процессы анализа, систематизации и управления текстовой информацией.

Предмет исследования – модели, методы и алгоритмы автоматизации процессов анализа, управления и систематизации специализированной текстовой информации.

Методы исследований. В процессе исследований использованы: методы машинного обучения и предварительной обработки текста; методы и алгоритмы анализа лингвистических особенностей языка; экспертный подход для решения поставленных задач управления и обработки информации; элементы аналитической алгебры и теории множеств.

Научная новизна полученных результатов:

1. Впервые предложена усовершенствованная общая модель автоматической систематизации и управления информацией, основанная на объединении достоинств синтагматических и парадигматических подходов.

Использование новой модели позволяет повысить полноту и точность работы модели в среднем на 32,5% и 31,5% соответственно.

2. Получила дальнейшее развитие и модернизирована модель классификации текстовой информации на основе внесения изменений в структуру алгоритма её построения, что позволяет повысить полноту и точность работы модели еще на 5,5% и 8,5 % соответственно.

3. Экспериментально обоснована модель вычислительной композиции распределения веса слов в текстовом документе. При разном распределении веса терма, повышение качества работы предложенной общей модели систематизации и управления информацией варьируется в пределах 10% в зависимости от используемой композиции.

4. Обосновано решение задачи повышения эффективности предложенной общей модели систематизации и управления текстовой информацией на основе разработанных модернизированных моделей классификации информации за счет внесения изменений в структуру алгоритма её построения и вычислительных композиций распределения веса слов в документах, правил отбора неинформативных признаков и способов взвешивания термов. Применение предложенных усовершенствований обеспечивает дополнительное повышение качества распределения информации на 27,5%.

Теоретическая значимость работы. Предложенная комплексная методика построения модели автоматической классификации и статистического анализа является совершенствованием существующих подходов к обработке информации и в дальнейшем может быть расширена и дополнена функциями автоматического и автоматизированного тематического анализа потоков текстовой информации для расширения количества тематик, по которым распределяются текстовые документы, а также повышением качества модели автоматической обработки информации. Структура статистических баз данных, формируемых с помощью предложенной технологии, позволяет ставить и решать большой спектр статистических и математических расчетных задач, и задач, связанных с принятием решений, имеющих место в информационных системах. Развитие

данной разработки может осуществляться путем дополнения ее новыми решениями в области морфологического, синтаксического и семантического анализа языков, для усовершенствования методов управления и систематизации специализированной текстовой информации.

Практическая значимость полученных результатов. Результаты исследований имеют широкий спектр применения для различных предметных областей. Предложенная практическая реализация усовершенствованной модели систематизации и управления информацией позволяет формировать текстовые базы данных классифицированной информации в автоматическом режиме. На основании результатов классификации имеется возможность формировать аналитические задачи и статистические базы данных по результатам обработки текстов, автоматизировать работу специалистов–аналитиков, осуществляющих тематический анализ текстовой информации, и ведение аналитических задач в различных предметных областях, что может послужить функциональным дополнением и развитием информационных систем различных организаций.

Практическое значение полученных результатов подтверждается:

– внедрением в практику организации информационных массивов и баз данных с целью совершенствования компьютерной технологии прогноза в отделе сдвижения земной поверхности и охраны подрабатываемых объектов (СЗПО) Республиканского академического научно–исследовательского и проектно–конструкторского института горной геологии, геомеханики, геофизики и маркшейдерского дела (РАНИМИ) (справка о внедрении № 01/140 от 20.05.2019 г.);

– внедрением в учебный процесс ГОУВПО «ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ» (справка № 52.1–05/19 от 14.05.2019 об использовании в учебном процессе при чтении лекций и проведении практических занятий на кафедрах «Искусственный интеллект и системный анализ» и «Прикладная математика» по дисциплинам: «Организация баз данных и знаний», «Информационные системы и технологии»,

«Стандартизация и сертификация в сфере информационных технологий», «Распределённые информационно–аналитические системы», «Корпоративные информационные системы»).

Методология исследования. В процессе исследования выполнялся анализ закономерностей морфологии естественного языка (русского), анализ структуры существующих словарей, поисковых запросов и логических моделей возможных запросов, математический анализ методики оценки релевантности качества классификации, применялись методы концептуального анализа и управления в системах автоматической систематизации специализированной информации и оценки эффективности их работы, современные методы автоматического анализа и управления текстовыми документами.

Положения, выносимые на защиту:

– обоснована новая модель автоматической систематизации и управления информацией, основанная на объединении достоинств синтагматических и парадигматических подходов, использование которой позволяет повысить полноту и точность работы модели в среднем на 32,5% и 31,5% соответственно;

– применение вычислительной композиции методов в модели распределения веса слов в тексте позволяет повысить качество работы модели в среднем на 10% в зависимости от используемой композиции;

– усовершенствованная конфигурация модели систематизации информации на основе правил отбора неинформативных признаков и способов взвешивания термов за счет внесения изменений в структуру алгоритма её построения и вычислительных композиций распределения веса слов в документах обеспечивает дополнительное повышение качества распределения информации на 27,5%.

По направлению исследований, содержанию научных положений и выводов, существу полученных результатов диссертационная работа

соответствует паспорту специальности 05.13.01 – Системный анализ, управление и обработка информации (по отраслям) согласно разделам:

2. Формализация и постановка задач системного анализа, оптимизации, управления, принятия решений и обработки информации.

4. Разработка методов и алгоритмов решения задач системного анализа, оптимизации, управления, принятия решений и обработки информации.

6. Методы идентификации систем управления на основе ретроспективной, текущей и экспертной информации.

Степень достоверности и апробация результатов обеспечивается полнотой анализа теоретических и практических исследований, положительной оценкой на научных конференциях и семинарах, выполненными публикациями.

Основные результаты настоящего диссертационного исследования представлены в 9 публикациях, в том числе 5 статей в изданиях, рекомендованных ВАК ДНР, основные положения и научные результаты диссертационной работы докладывались, обсуждались и получили положительную оценку на следующих 4 конференциях:

– VIII Международная научно–техническая конференция «Информационные перспективы Донбасса» (г. Донецк, 2017);

– V Международная научно–техническая конференции «Современные информационные технологии в образовании и научных исследованиях» (г. Донецк, 2017);

– XXV международная научно–техническая конференция «Машиностроение и техносфера XXI века» (г. Севастополь, 2018);

– Научно–техническая конференция «Донецк будущего глазами молодых ученых» (г. Донецк, 2018).

Личный вклад. Основные научные результаты диссертации включают в себя разработку новой модели систематизации и управления информацией на основе объединения парадигматических и синтагматических подходов, комплекс вычислительных композиций в модели распределения веса слов в тексте для

предсказания релевантной для документа тематики или набора тематик, а так же усовершенствованную конфигурацию модели систематизации информации на основе модернизации алгоритма её построения и вычислительных композиций распределения веса слов в документах, применение которых повышает качество и скорость автоматической систематизации и обработки информации.

Все выносимые на защиту положения получены автором лично.

Публикации. Основные научные результаты диссертации опубликованы автором самостоятельно и в соавторстве в 9 научных изданиях, 5 из них в рецензируемых научных изданиях: в том числе 2 – в рецензируемых научных журналах и изданиях, в которых должны быть опубликованы основные научные результаты диссертаций на соискание ученой степени кандидата и доктора наук в Российской Федерации, и 3 - в специализированных научных изданиях, рекомендованных ВАК ДНР, 4 – по материалам научных конференций.

Объем и структура диссертации. Диссертация изложена на 161 странице машинописного текста и состоит из введения, четырех глав, заключения, списка литературы, приложений. Работа иллюстрирована 28 рисунками, содержит 19 таблиц. Указатель литературы включает 140 источников.

ГЛАВА 1
АВТОМАТИЗАЦИЯ УПРАВЛЕНИЯ ТЕКСТОВОЙ ИНФОРМАЦИЕЙ – АНАЛИЗ СУЩЕСТВУЮЩИХ ПОДХОДОВ, ТЕНДЕНЦИЙ И ЗАКОНОМЕРНОСТЕЙ УПРАВЛЕНИЯ И СИСТЕМАТИЗАЦИИ ТЕКСТОВОЙ ИНФОРМАЦИИ

В большинстве организаций значительная часть полезных знаний содержится в документальных базах данных, а темпы роста, содержащегося в них количества документов, постоянно увеличивается. Своевременное получение знаний в автоматическом режиме затрудняется слабой упорядоченностью текстов на естественном языке. Такие знания могут быть извлечены экспертом, но с учетом огромного числа электронных документов их эффективная обработка человеком становится весьма затратной как по времени, так и по используемым ресурсам. А отсутствие возможности вовремя и быстро получить необходимую информацию по нужной теме сделает бесполезной большую часть накопленных знаний, как следствие появляется необходимость в управлении, анализе и систематизации текстовой информации.

Анализ и систематизация данных предполагает работу с информацией, ее глубоким осмыслением и принятием адекватных решений по управлению ею. Под систематизацией понимаются объекты, организующиеся в некую систему, на базе выбранного принципа, а система представляет собой совокупность разнородных элементов, предназначенных для достижения поставленной цели.

Основной функцией систем извлечения знаний является информационный поиск полезных сведений в документальных базах. Более точное и быстрое извлечение знаний имеет своей конечной целью информационную поддержку эксперта или автоматизированной системы при принятии решения в поставленной задаче (вопросе). Так изучение процедур принятия решений и организация системы составляет актуальную проблему создания и эксплуатации сложной системы. Сложная система включает в себя системы с большим числом элементов различного типа и с разнородными связями между ними. Связь

обеспечивает возникновение и сохранение целостных свойств системы. Она определяется как ограничение степени свободы элементов. Под элементами понимается предел членения системы. Установлением систематизированных связей между элементами исследуемой системы занимается системный анализ. Системный анализ и управление информацией подразумевает под собой совокупность процедур, системных идей, подходов, теорий и методов, предназначенных для анализа объектов и процессов как систем [10].

Применение системного анализа для построения такой системы означает применение специально разработанных приемов, методик, типовых моделей организации системы и принятия ращений. Рассмотрим основные результаты выполненного в исследовании анализа существующих подходов, тенденций и закономерностей управления и систематизации текстовой информации.

1.1 Управления информацией в условиях ее актуализации, тенденции роста информации

Появление новых технологий способствует научному прогрессу. Тенденция увеличения объемов и распространения информации в электронном виде стимулирует активное развитие автоматических систем обработки информации. В большинстве организаций значительная часть полезной информации содержится в электронных базах данных.

Организации в ходе своего существования формируют достаточно большие архивы документации. В данных архивах содержатся не только результаты официального документооборота (распоряжения, приказы и пр.), но и техническая документация по выполненным и текущим проектам: планы, технические отчеты, проектная документация и т.д.

Значительная часть хранящейся информации оформляется в виде текстового описания, анализ и систематизация которой предполагает ее глубокое осмысление, работу с данными, принятием адекватных решений относительно анализа и управления той или иной ситуацией:

- получение дополнительной информации;
- анализ имеющейся в наличии информации, относящейся к анализируемой проблеме;
- тематическую обработку информации;
- визуализацию и подготовку аналитических отчетов, их верификацию;
- принятие управленческих решений на основе новых знаний.

Большая часть используемых документов формируется в виде текстового описания. В связи с ростом используемых персональных компьютеров в организациях чрезвычайно быстро растет и количество текстовой информации на естественном языке хранящейся в электронном виде.

Статистика увеличения объема данных, создаваемых на протяжении последних лет, поражает воображение. В 2013 г. количество информации, хранящейся в мире составило 1,2 зеттабайта, что приравнивается к 1,2 млн. петабайт или 1,2 трлн. гигабайт [11]. Так по прогнозам компании специализирующейся на аналитике, в сфере информационных технологий, а именно IDC, общее количество информации будет удваиваться каждые 2 года. Как следствие к 2020 г. достигнет порядка 40 зеттабайта, это говорит о том, что на каждого жителя Земли будет приходиться по 5200 ГБ данных в соответствии с рисунком 1.1 [12].

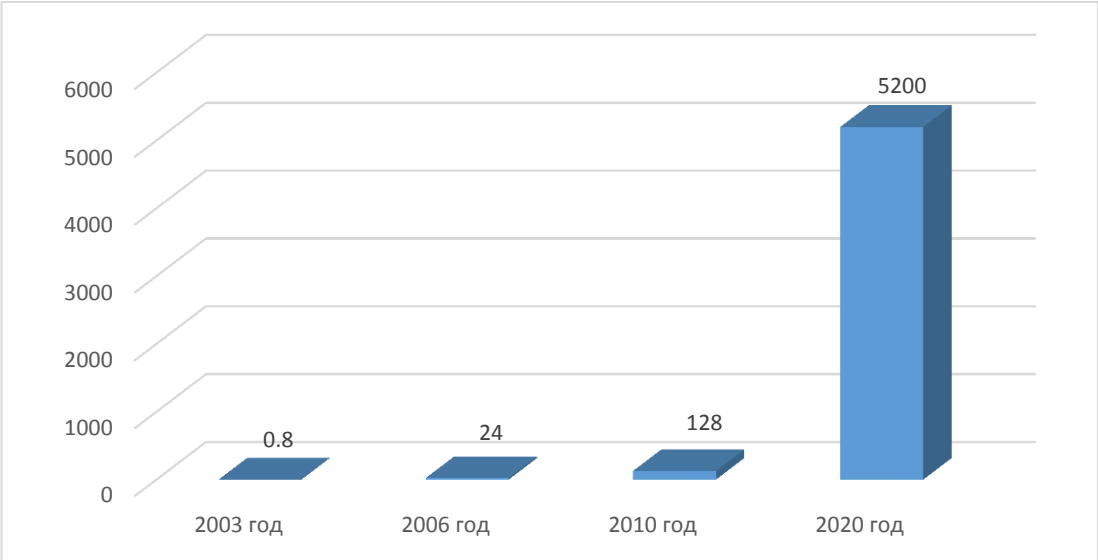

Рисунок 1.1 – Прогноз количества ГБ информации в мире на человека

Таким образом, рост информации делает задачи управления и систематизации текста актуальными для проведения исследований и разработки средств обработки текстовой информации, которая позволит повысить качество и облегчить работу эксперта.

1.2 Задача анализа и степень разработанности технологий автоматической обработки информации

С ростом объема информационного потока специалистам–аналитикам становится все труднее заниматься тематическим анализом информации и ведением аналитических задач вручную при существующих средствах автоматизации. В соответствии с оценками экспертов около 70% используемых и накопленных цифровых данных находится в несистематизированной текстовой форме и лишь 30% образуют другие виды данных. Экспоненциальное с течением времени увеличение количества несистематизированных данных приведет по существу к коллапсу традиционной системы распределения и получения текстовой информации. Как следствие рутинная операция анализа и поиска необходимых данных превращается в малоэффективный и трудоемкий процесс, вызывающий информационную перегрузку пользователей. В данной ситуации особую актуальность приобретают работы по созданию систем автоматически систематизирующих и управляющих текстовой информацией. Как показывает практика, результаты распределения по предметным областям документом «вручную», то есть путем экспертного отнесения к имеющейся тематике, как правило, не превышает 80% [1 с. 5].

Классификация текса является одним из способов систематизации данных, которая подразумевает распределение текстовых документов по заранее определенным категориям [13]. На стыке двух областей находятся методы классификации текстовых документов, а именно машинного обучения и информационного поиска. Объединяет данные области средства оценки качества классификации информации и средства представления документов. Различия

заключаются лишь в способах отнесения документа к соответствующей тематике. Точность используемых методов классификации существенно зависит от выполнения априорных допущений и предположений, также стоит учитывать структуру текстовых данных, например, количество классов, вид «пограничной» области между классами, однородность и размеры классов. Программные разработки, обесточивающие автоматическую классификацию информационных массивов, существуют [14 с. 39], но они, как правило, лишь частично решают проблему автоматической систематизации и управления информацией или ведения аналитических задач. В этой связи возникает необходимость в решении задач создания усовершенствованной системы анализа информации, состоящей в управлении и систематизации текстовых документов.

При построении системы систематизирующей текстовую информацию необходимо учесть следующие факторы [15]:

- количество информативных, то есть полезных для классификации терминов, или признаков, которые как правило существенно превосходят количество имеющихся в выборке документов, что затрудняет определение наилучших оценок параметров и обучение методов;
- чрезвычайно велик объем вычислительных операций, используемых для анализа и обработки текстовых документов, что делает процесс классификации крайне трудоемким и дорогостоящим;
- получаемая матрица «документ – термин» оказывается сильно разреженной, так как большое число терминов встречается только в нескольких или в одном документе;
- несистематизированная информация не имеет общепринятых правил и единого текстового формата, что делает анализ и обработку документов практически невозможным без разработки комплексной модели процесса обрабатывающего текстовую информацию, в отличие от систематизированной информации, которая, как правило содержит фактические сведения, представленные числами.

Для создания эффективных систем управления и систематизации информацией требуется совместное применение методов классификации на основе машинного обучения и методов, основанных на знаниях. Поскольку содержимое и состав анализируемых документов постоянно изменяется, и одним из направлений адаптации к этой динамике является использование данных методов. Целью методов машинного обучения для задачи классификации текстовых документов является построение модели классификации, основанной на обучающем наборе, а далее применении построенной модели для предсказания набора или одного класса, релевантных для нового документа. Представленный подход обеспечивает качество классификации, сравнимое с качеством классификации, производимой человеком.

Тестированию и разработке алгоритмов данного вида, а также связанными с ними алгоритмами представления текстов в настоящее время посвящены труды таких авторов как Агеев М. С., Joachims T., Кураленок И. Е., Scbastiani F., Schapire R. E., Schutze H. и другие [16, 17, 18]. Стоит отметить, что в современных прикладных задачах обучающие наборы имеют достаточно большой размер, ввиду чего интерес представляет разработка эффективных методов машинного обучения.

В задачах текстовой классификации лучше всего зарекомендовали себя метод опорных векторов и метод построения алгоритмических композиций на основе бустинга (улучшения). Анализ российских и зарубежных публикаций демонстрирует, что основные усилия исследователей [19, 20] направлены на построении классификаторов, обладающих высокими показателями точности и полноты. Стоит учитывать, что при разработке методов классифицирующих текстовую информацию, которая имеет высокую размерность, большое число терминов, описывающих документ, отдельное внимание требуют также вопросы быстродействия, то есть уменьшение времени, затрачиваемого на отнесение документа к одному из классов. В литературе практически нет работ, посвященных проблемам производительности классификаторов, как следствие,

проблема быстродействия классификации ложатся на плечи разработчиков систем машинного обучения [21].

При решении практических задач реализация мер, направленных на повышение точности классификации, как правило, приводит к снижению быстродействия. Гарантия высокого быстродействия в крупных поисковых системах являешься особо важным свойством, для решения задач анализа поисковых запросов, предоставляемых пользователем в режиме реального времени, приоритизация URL (Uniform Resource Locator) адресов web страниц [22], число которых движется в направлении нескольких миллиардов, для их загрузки в поисковые машины. Стоит учитывать, что подобные системы принадлежат к классу высоконагруженных, то есть обладают либо большим объемом данных, либо большим количеством одновременных сессий пользователей, или совокупностью данных критериев. Для решения определенной задачи качество и быстродействие работы являются ключевыми факторами, влияющими на выбор метода, используемого для таких систем [23].

Таким образом, для построения системы является актуальным проведение исследований и разработка программных средств, классификации текстовых документов на основе методов машинного обучения, обеспечивающих высокое быстродействие при сохранении или повышении качества (полноты и точности) классификации.

1.3 Анализ технологий автоматической обработки текста и методология моделирования сложных систем

Подраздел «классификация текстов» обработки естественных языков (англ. Natural Language Processing) в последнее десятилетие развивается достаточно интенсивно [24].

Несмотря на то, что начало исследований в области искусственного интеллекта произошло еще в 1959 году, когда Артур Самуэль (Arthur Samuel), изобретатель и исследователь первой самообучающейся компьютерной

программы игры в шашки, ввел в научный обиход термин «машинное обучение». Результат его работы заключается в создании первой программы игры в шашки. Самуэль определил машинное обучение как «процесс, в результате которого компьютеры способны показать поведение, которое в них не было явно запрограммировано». Одной из первых программ, с реализованными функциями самообучения и наглядной демонстрацией базовых принципов искусственного интеллекта стала «Checkers–playing» [25].

С промежутком почти в десятилетие в 1967 был предложен метрический алгоритм классификации, другими словами метод k ближайших соседей. Алгоритм которого позволил компьютерам использовать простые шаблоны распознавания [26].

Прежде чем Джеральд Дежонг (Gerald Dejong) представит концепцию, основанную на обучении (Explanation Based Learning) пройдет еще десятилетие, концепция была представлена в 1981 году. Следом за ним в 1985 году Терри Сейновски (Terry Sejnowski) создает NetTalk искусственную нейронную сеть.

Дэвидом Румельхартом (David Rumelhart) и Робби Вильямсом (Robbie Williams) был заново открыт и популяризирован в 1986 году алгоритм обратного распространения ошибки. Этот алгоритм также был получен другими учеными независимо друг от друга. Впервые он был предложен Полом Вербосом (Paul Werbos) в 1974 году [14 с. 16].

Джеффри Хинтон (Geoffrey Hinton), ученый в области искусственных нейросетей, в 2006 ввел в обиход термин «Глубинное обучение» (Deep learning).

Суперкомпьютер IBM Watson, оснащенный системой искусственного интеллекта был создан в 2011 году. Он одержал победу в телевикторине Jeopardy. Его соперниками были маститые игроки Брэд Раттер (Bred Ratter) и Кен Дженнингс (Ken Jennings) [27].

И лишь в 2012 Google запускает облачный сервис Google Prediction API для машинного обучения, помогающий анализировать несистематизированные данные [28].

Следом за ним в 2015 Amazon запустила собственную платформу машинного обучения – Amazon Machine Learning. В этом же году Microsoft создает платформу Distributed Learning Machine Toolkit, которая предназначена для децентрализованного машинного обучения [29].

Существующие алгоритмы классификации возможно применять не только для классификации текстовой информации, но также и для извлечения из них дополнительной информации. Наличие у системы функции, которой нет у составляющих ее частей и разных по типу и многочисленных связей между отдельно существующими элементами системы, является особенностью, проявляющейся при построении таких сложных систем. Характеристики взаимодействий между элементами сложной системы определяются направленностью на выполнение функции системы, внутренними свойствами и определенным порядком [30].

В настоящее время, для описания сложных систем используют существующее множество методологий: *ARIS; IDEF0; UML; IDEF3; DFD; WORKFLOW* и другие; или инструментальные средства такие как: *ERWin*; *PowerDesigner*; *BPWin* и другие [31, 32].

В объектно–ориентированном и структурном анализах применяются средства, моделирующие в форме диаграмм определенного вида деловые отношения и процессы между данными в системе. Данные средства соответствуют определенным видам системных моделей, наибольшее распространение среди которых получили следующие [33 с. 11; 34; 35]:

- *Integrated Definition* (*IDEF)* – семейство структурных моделей и соответствующих им диаграмм;
- *DFD* (*Data Flow Diagrams*) – диаграммы потоков данных;
- *ERD* (*Entity-Relationship Diagrams*) – диаграммы «сущность–связь»;
- *Workflow* – технология управления потоками работ;
- *BPMN (Business Process Modeling Notation*);

- *CPN* (*Color Petri Nets*) – средства имитационного моделирования, основанные на математическом аппарате раскрашенных сетей;
- объектно-ориентированные методологии на основе унифицированного языка моделирования *UML*;
- интегрированные средства и методологии широкого назначения, например, *ARIS*.

Потребность в построении сложных моделей и развитии технологий во многом связано с тем, что объём информации, хранимой на электронных носителях, с каждым годом значительно возрастает, как следствие возникает необходимость в эффективных алгоритмах, предназначенных для анализа и обработки документов, созданных на естественном языке. Усовершенствование алгоритмов, в свою очередь, становится возможным благодаря увеличению производительности и мощности современных компьютеров [36, 37].

Синтаксис языка представляет собой набор структурных компонентов языка, а также правил и характеристик, определяющих связи между компонентами языка.

Для решения задач моделирования сложных систем используются – методологии семейства *ICAM* (*Integrated Computer – Aided Manufacturing*) – *IDEF*, использование которых предоставляет возможность анализировать и отображать в различных разрезах модели деятельности широкого спектра сложных систем. При этом глубина и широта обследования процессов в системе определяются самим разработчиком, что предоставляет возможность не перегрузить создаваемую модель излишними данными. Правила, блоки, диаграммы и стрелки являются компонентами синтаксиса *IDEF0*. Блоки демонстрируют функции, определяемые как преобразование, деятельность, действие, процесс или операция. Стрелки демонстрируют материальные объекты или данные, связанные с функциями. Необходимость применения компонент определяют правила, а формат словесного и графического описания моделей обеспечивают диаграммы. Основу для управления конфигурацией модели образует формат [38].

IDEF – методологии создавались в рамках программы компьютеризации промышленности – *ICAM*, в ходе реализации которой выявилась потребность в разработке методов анализа процессов взаимодействия в производственных системах. Принципиальным требованием, при разработке рассматриваемого семейства методологий, была возможность эффективного обмена информацией между всеми специалистами – участниками программы *ICAM* (отсюда название: *Icam DEFinition – IDEF* другой вариант – *Integrated DEFinition*). После опубликования стандарта он был успешно применен в самых различных областях, показав себя эффективным средством анализа, конструирования и отображения процессов [33 с. 9].

К семейству *IDEF* в настоящий момент в соответствии с рисунком 1.2 можно отнести следующие стандарты [33 с. 12]:

IDEF0 – Function Modeling – «технология функционального моделирования сложных систем». Используя наглядный графический язык *IDEF0*, исследуемая система появляется перед аналитиками и разработчиками в виде набора взаимосвязанных функциональных блоков. Первый этап изучения любой системы заключается в построении ее модели средствами *IDEF0*. Методология *IDEF0* является следующим этапом развития известного графического языка описания функциональных систем *SADT* (*Structured Analysis and Design Technique*) [33 с. 12].

IDEF1 – Information Modeling – «технология моделирования информационных потоков внутри системы». Использование которой предоставляет возможность анализировать и отображать взаимосвязи и структуру потоков информации [33 с. 12].

IDEF1X (*IDEF1 Extended*) – *Data Modeling* – технология построения реляционных структур, то есть баз данных, состоит в построении инфологических моделей данных типа «сущность-связь» (*ERD – Entity-Relationship Diagram*). Обычно, используется для моделирования реляционных баз данных, имеющих непосредственное отношение к рассматриваемой системе [39 с. 14].

IDEF2 – Simulation Model Design (проектирование модели поведения) – «технология динамического моделирования систем». Развитие этого стандарта приостановилось на самом начальном этапе из-за достаточно серьезных сложностей анализа динамических систем. Сейчас существуют алгоритмы и их компьютерные реализации, предоставляющие возможность преобразовывать набор статических диаграмм *IDEF0* в динамические модели, созданные на базе «раскрашенных сетей Петри» (*CPN – Color Petri Nets*) [33 с. 12].

IDEF3 – Process Description Capture (сбор данных по описанию процессов) – «технология документирования процессов, происходящих в системе». При построении и исследовании технологических процессов на предприятиях и в организациях используются подобные системы. Последовательность операций и сценарии для каждого процесса описываются, посредствам использования стандарт *IDEF3*, который имеет прямую взаимосвязь с технологией *IDEF0*, причем каждая функция *IDEF0* может быть отображена при помощи отдельного процесса *IDEF3* [33 с. 9].

IDEF4 – Object–Oriented Design – «технология построения объектно-ориентированных систем». Использование средства *IDEF4* позволяет наглядно демонстрировать структуру объектов, а также заложенные принципы их взаимодействия, чем позволяет оптимизировать и анализировать сложные объектно-ориентированные системы. Имеет связь с технологией *UML* [33 с. 12].

IDEF5 – Ontology Description Capture – «технология онтологического исследования сложных систем». Определение терминов, относящихся к какой-либо предметной области, является онтологией. Набор правил и использования определенного словаря терминов позволит описать онтологию системы технологи *IDEF5*. На основании которых, в некоторый момент времени, могут быть сформированы достоверные утверждения о состоянии рассматриваемой системы. Выводы о дальнейшей оптимизации и развитии системы формируются на основании данных утверждений [33 с. 12].

IDEF6 – Design Rational Capture – «технология использования рационального опыта проектирования». Ее назначение заключается в сохранении

рационального опыта проектирования информационных систем используемых для предотвращения структурных ошибок при новом проектировании.

IDEF7 – Information System Auditing – «технология аудита информационной системы».

IDEF8 – User Interface Modeling – «технология проектирования интерфейса пользователя». Применение *IDFE8* фокусирует внимание разработчиков интерфейса на программировании желаемого взаимного поведения пользователя и интерфейса на трех уровнях [33 с. 12]:

- выполняемой операции, что представляет собой данная операция;
- сценарии взаимодействия, определение которого осуществляется специфической ролью пользователя, то есть по какому сценарию она должна выполняться тем или иным пользователем;
- на деталях интерфейса, другими словами, какие элементы управления рекомендует интерфейс для выполнения операции.

IDEF9 – Scenario – Driven IS Design – технология анализа имеющихся ограничений и условий, в том числе политических, юридических, физических и их влияния на принимаемые решения в процессе вторичного проектирования.

IDEF10 – Implementation Architecture Modeling – «моделирование архитектуры выполнения, то есть физической реализации системы» [33 с. 12].

IDEF11 – Information Artifact Modeling – «информационное моделирование артефактов».

IDEF12 – Organization Modeling – «организационное моделирование».

IDEF13 – Three Schema Mapping Design – «трехсхемное проектирование карт».

IDEF14 – Network Design – «технология моделирования компьютерных сетей». Технология предоставляет возможность выполнять анализ и представление данных при проектировании компьютерных сетей на графическом языке с описанием требований к надежности конфигураций, сетевых компонентов, очередей и тому подобное [33 с. 12].

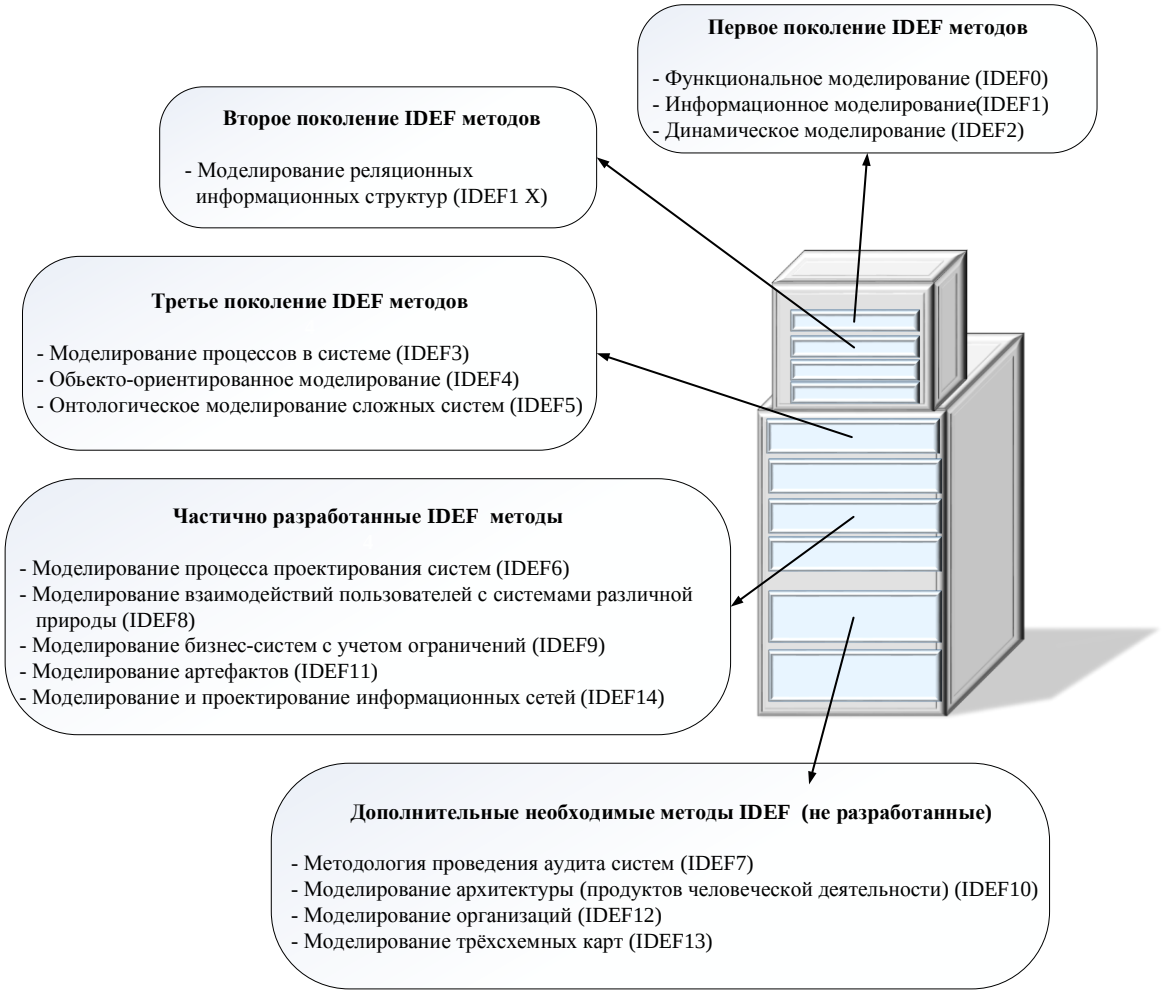

Рисунок 1.2 – Семейство стандартов структурного моделирования *IDEF*

Стандарты технологий *IDEF13*, *IDEF12*, *IDEF11*, *IDEF10*, *IDEF7* определяются как востребованные, но так и не были до конца разработаны [33 с. 15].

При использовании методологий семейства *IDEF*, появляется возможность эффективного анализа и отображения модели деятельности широкого спектра сложных систем. Наибольшее применение и распространение получили методологии *IDEF1* (*IDEF1X*) и *IDEF0* [39 с. 14].

1.3.1 Процесс построения *IDEF0*–модели

Прежде чем начать построение любой модели важно определить направление модели, а именно ее цель, точку зрения и контекст.

Цель используется для определения назначения модели и выявления причины ее создания: инструмент проектирования, функциональная спецификация и так далее.

Точка зрения способствует определению того, что может быть «видно» в рамках контекста с определенной «точки зрения» или перспективы. Принятие различных точек зрения зависит от цели, которая подчеркивает многогранность объекта, но в одной модели всегда используется только одна точка зрения [40]. Контекст определяет смысл модели, как части окружающей среды, что определяет границу со средой путем представления внешних интерфейсов - дуг. Контекст модели устанавливает контекстная диаграмма.

Ограничение контекста определяется как отправная точка для любого анализа. У аналитика возникает необходимость в принятии решения, о том, что будет главным, центральным элементом, до того, как будет создан самый верхний блок. Каждый следующий шаг необходимо сверять с начальной целью, а те данные, которые ей не соответствуют, откладываются для следующего моделирования.

В начале моделирования создаются диаграммы *A–0* в соответствии с рисунком 1.3. Далее изображается одиночный блок, хранящий название функции, которая охватывает все возможности, контекста представляемой системы, с использованием выходных, управляющих и входных дуг для представления объектов и данных системы, реализующих интерфейс с окружающей ее средой [41].

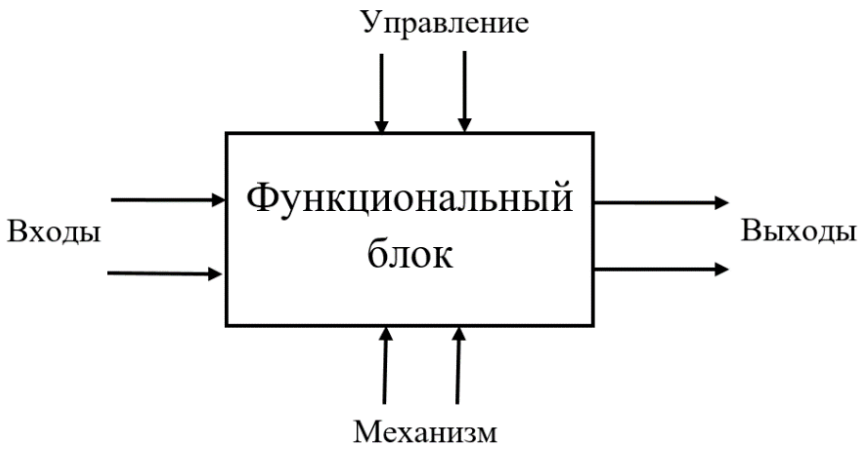

Рисунок 1.3 – Общий вид диаграммы *A–0*

Данная одноблочная диаграмма обуславливает контекст для полной модели и формирует основания для дальнейшей декомпозиции. Точка зрения и цель фиксируются на контекстной диаграмме *A–0*.

1.3.2 Принципы моделирования в *IDEF0*

Три базовых принципа моделирования процессов реализованы в *IDEF0*:

- принцип функциональной декомпозиции;
- принцип ограничения сложности;
- принцип контекста.

Способ моделирования типовой ситуации, то есть когда любая функция, операция, действие может быть разбита, декомпозирована, на гораздо более простые функции, операции, действия, это то, что представляет собой принцип функциональной декомпозиции. Иначе говоря, сложная организационная функция представляется при помощи совокупности элементарных функций. Заглянуть вовнутрь блока и детально рассмотреть состав и структуру функции позволяет использование фракции графики, в виде блоков в соответствии с рисунком 1.4 [42].

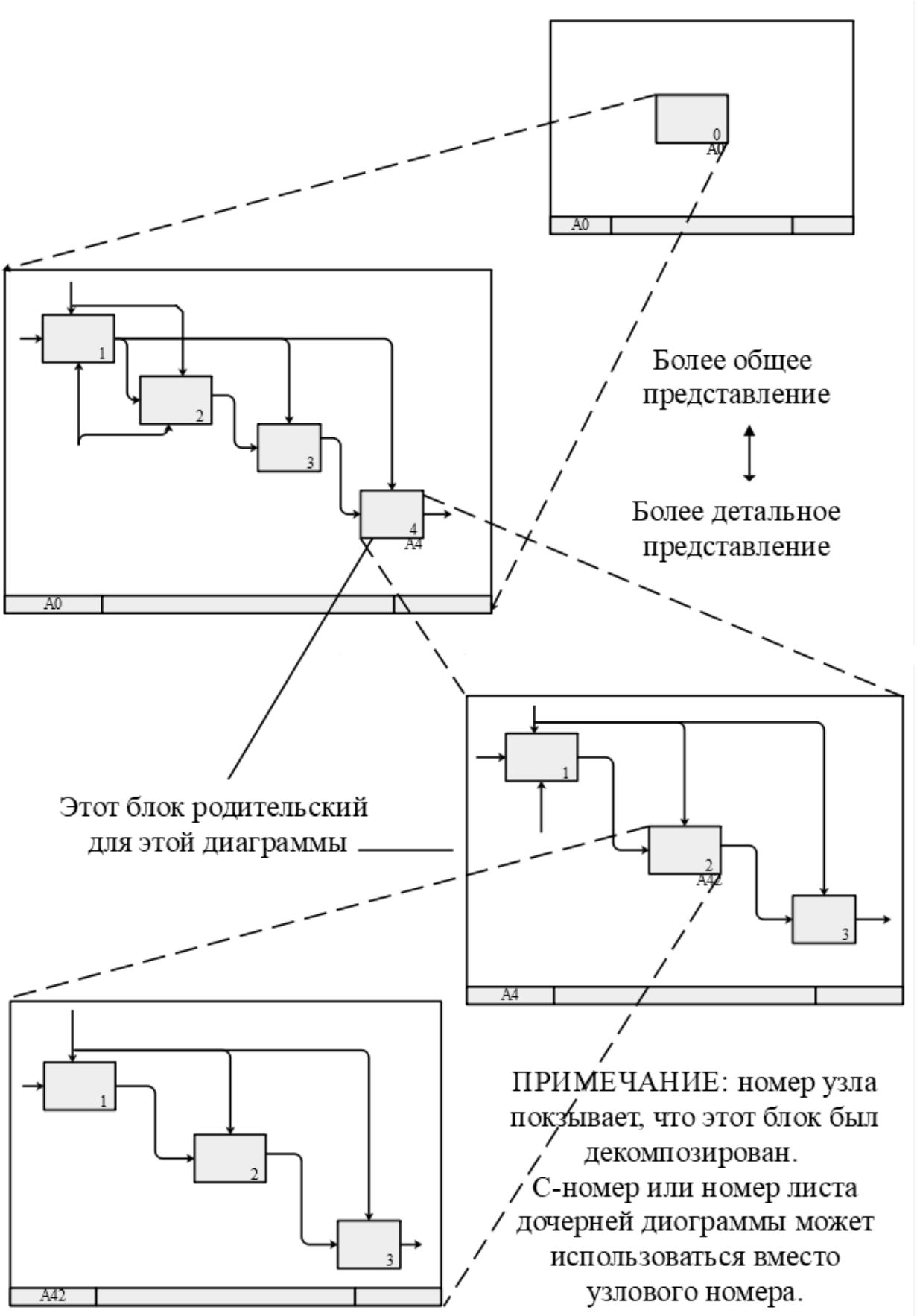

Рисунок 1.4 – Принцип функциональной декомпозиции

Существенное условие удобочитаемости и разборчивости при работе с *IDEF0* заключается в принципе ограничения сложности. Суть данного блока заключается в том, чтобы количество используемых блоков, представленных на диаграмме, было не более шести и не менее двух. Следствием соблюдения данного принципа является получение понятных и легко поддающихся анализу функциональных процессов, хорошо систематизированных и представленных в виде *IDEF0* модели [33 с. 9].

Принцип моделирования контекстной диаграммы делового процесса начинается с построения контекстной диаграммы. На данной диаграмме отображён только один блок, представляющий главную функцию моделируемой системы. Которая представляет «миссию» системы, ее значимость в окружающем мире. Невозможно корректно сформулировать главную функцию, не имея представления о стратегии [43].

При формировании главной функции требуется всегда иметь в виду точку зрения на модель и цель моделирования. Одна и та же организация может быть представлена по–разному, в зависимости от того, с какой точки зрения её рассматривают, например, налоговой инспектор и директор предприятия видят организацию совершенно по–разному.

Контекстная диаграмма «фиксирует» границы моделируемой организационной системы, другими словами определяет то, как моделируемая система взаимодействует со своим окружением, что достигается за счет изображения дуг, соединенных с блоком, демонстрирующих главную функцию [44].

1.4 Целенаправленные системы и управление

В завершающей части данной главы продемонстрируем понятия, относящиеся к постановке перед системой некоторой сформулированной цели. Искусственные системы почти всегда определяются, как целенаправленные [45].

Понятие цели системы определяется, как задача достижения желаемого состояния или получения ожидаемого выходного воздействия системы [46].

Стоит обратить внимание на то, что двоякая трактовка цели – через состояние системы или через выходное воздействие – удобна в приложениях. Теоретически есть возможность считать цель только выходным воздействием, а желаемое состояние ввести в список данных воздействий. В частных случаях такая интерпретация состояния системы имеет возможность вносить дополнительные сложности, а в отдельных случаях приводит к неразберихе.

Постановка цели для системы в целом, то есть глобальной цели, приводит к потребности:

- в формулировке локальных целей, стоящих перед группами элементов и элементами системы;
- в целенаправленном вмешательстве в функционирование, создание и строение системы.

Обе данные операции плотно связаны, однако с точки зрения решения практических задач обычно сначала разделяют глобальную цель на набор локальных, а затем ищут средства достижения локальных целей [47].

Многоуровневое иерархическое строение, как правило, имеет набор локальных целей, которое в некоторой степени соответствует общей иерархии системы. В данном случае понятие «локальные цели» подразумевает собирательный термин для целей всех иерархических уровней. Для каждой цели есть возможность указания, к какой цели более высокого уровня она относится, исключением являются цели самого низшего уровня. Используя модульное строение системы, локальные цели представляются как требования к выходам, входам и характеристикам модулей. Благодаря продуманным требованиям на входах согласовывают модули таким образом, что состоящая из них система выполняет глобальную цель в соответствии с рисунком 1.5. Как следствие, локальные цели выступают важным регулятором организации элементов и частей в целенаправленную систему, а их скоординированность направляет проводимые в системе изменения в единое русло [48].

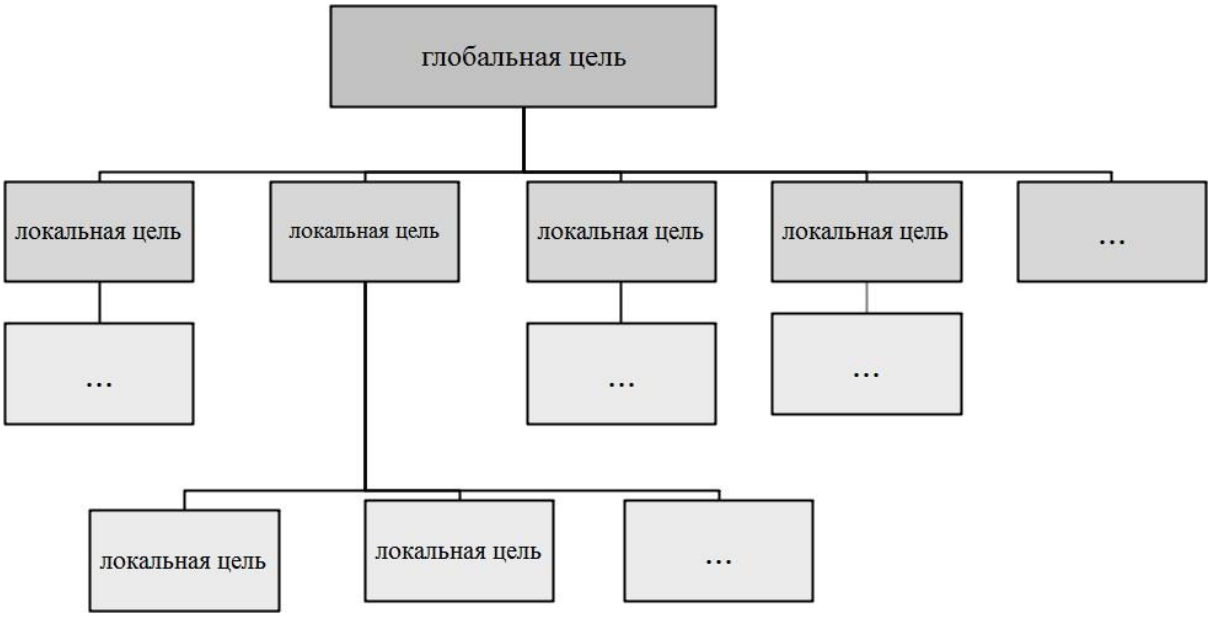

Рисунок 1.5 – Иерархическое многоуровневое строение системы

Согласование, как правило, является сложной, тяжело формализуемой процедурой. При этом определенная локальная цель может производиться и таким образом, что усложнит исполнение смежной цели, и только компромиссное решение между ними обеспечит движение к глобальной цели системы.

Целенаправленное вмешательство в процессы системы определим, как управление. Важнейшим понятием для целеустремленных систем является управление [49].

Управление является универсальным термином с огромным многообразием его конкретных реализаций [50 с. 56]:

- в математических моделях можно выбирать графовые структуры, алгоритмы, функции, числа;
- в технических системах – различные сигналы, включая команды ЭВМ, геометрические размеры, силы;
- физические величины – перемещение и концентрация веществ, от жесткости до температуры материала;
- в экономике – расстановку кадров, материальные ресурсы и сроки их поставки, размеры финансирования;

– в социальной области – организацию новых коллективов, влияние на общественное мнение, действия, советы, приказы [50 с. 56].

Здесь представлена небольшая часть того, чем в целях управления можно оперировать в сложной системе [50 с. 55].

Для строгого подхода в управлении необходимо однозначное и четкое определение:

– того, чем распоряжаемся;
– каковы границы, в которых можно осуществлять выбор;
– каким образом данное управление влияет на процесс [51].

Но при решении реальных задач все перечисленные требования могут оказаться неточными, а два последних редко используются. Не соблюдение требования приводит к тому, что результат управления не приведет к поставленной цели. Если описание процесса в системе отсутствует, то положение возможно использовать и в строгой трактовке управления. В результате чего появляется опыт работы с «черным ящиком» в соответствии с рисунком 1.6 [52].

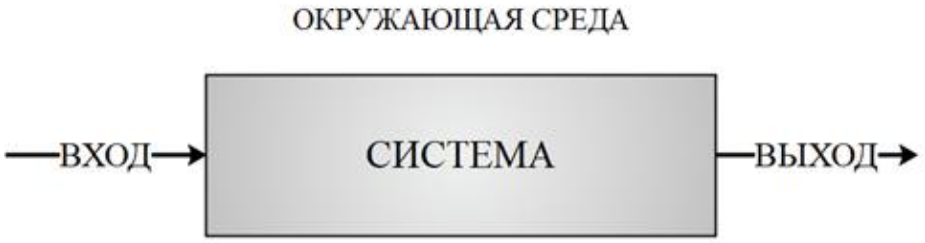

Рисунок 1.6 – Модель «черного ящика»

Исходя из цели, когда в ситуации, отсутствует управление обеспечивающее выполнение поставленной цели, возникает потребность в расширении пределов управления, путем внесения новых управляющих воздействий, использование которых иногда в корне меняют структуру системы. Таким образом, в возникшей ситуации область достижимости цели пропадает [53].

Анализ понятия управления является источником, формирующим управляющие команды:

- решения и действия человека, ответственного лица, эксперта администратора, диспетчера, водителя, оператора, и так далее;
- технические средства — это компенсирующие и стабилизирующие системы, управляющие и другие ЭВМ, программные устройства, микропроцессоры, регуляторы и так далее.

Перечисленные источник включают в себя ряд общих свойств, обладающих воздействующими характеристиками, влияющими на процессы в системе, но в них также имеются и существенные различия. Так как источники имеют как достоинства, так и недостатки, рекомендовано их совместное использование. Представленные объекты традиционно называют автоматизированными системами управления (АСУ). Современные тенденции развития АСУ определяются, как перепоручение техническим средствам формирования управляющих воздействий и выполнение сопутствующих операций, выполнение которых происходит быстрее и качественнее человека [33 с. 39].

Выполненный анализ материала позволил сформулировать цель и задачи.

Целью исследования является модернизация методик, моделей и алгоритмов системного анализа, искусственного интеллекта и компьютерной лингвистики для повышения эффективности автоматизации, систематизации специализированной текстовой информации больших объёмов с учетом особенностей русскоязычных текстовых документов.

Для достижения цели диссертационного исследования поставлены и решены задачи:

1. Проанализировать существующие методы, модели и алгоритмы, используемые для решения задач систематизации и управления специализированной текстовой информации.

2. Разработать модифицированную модель автоматической обработки специализированной информации, основываясь на объединении достоинств парадигматических и синтагматических подходов.

3. Выявить закономерности изменения качества работы модели обработки текстовой информации при внесении изменений в ее конфигурацию.

4. Осуществить исследование выполненной разработки усовершенствованной модели обработки текстовой информации.

5. Повысить эффективность применения предложенных моделей и методов для систем автоматической обработки специализированной информации.

6. Проанализировать качество работы предложенных средств, при решении практических задач классификации текстов.

1.5 Выводы к главе 1

В данной главе выполнен анализ основных подходов к моделированию и анализу слабо систематизированных систем.

1. Управление и систематизация по содержанию большого количества текстов является сложной актуальной задачей. Ее выполнение ограниченным количеством специалистов и затрачиваемым временем в условиях постоянного поступления новой информации практически невозможно.

2. Анализ современных публикаций позволяет утверждать, что существует значительный разрыв между методами систематизации и управления информацией, основанными на машинном обучении, и методами основанными на знаниях.

3. Выполнен анализ принципов построения систем управления, систематизации и обработки текстовой информации, изучены особенности их работы. Формализация единого универсального системного способа представления знаний позволяет создать соответствующие алгоритмы и инструментальные средства для обработки знаний различного типа единообразным способом и с помощью единого формального аппарата, построение которого осуществляется на основе методов и алгоритмов машинного обучения в рамках анализа специализированной текстовой информации.

4. Решения задачи создания единых основ представления накопленных знаний и управления ими осуществляется за счет интеграции и универсализации существующих способов систематизации таких знаний. Предлагается способ преобразования знаний, приведенных к единому виду, при помощи моделей в стандартах серии *IDEF*, выбор компонентов которой осуществляется среди возможных решений на основе специально разработанных приемов, методик и типовых моделей организации системы и принятия решений.

5. Показано, что разработка новой модели управления и систематизации специализированной текстовой информации, и модернизация применяемых в ней методик, методов и алгоритмов системного анализа, искусственного интеллекта и компьютерной лингвистики позволяет повысить эффективность управления, систематизации и обработки специализированной текстовой информации больших объёмов с учетом особенностей русскоязычных текстовых документов.

ГЛАВА 2
АНАЛИЗ И ВЫБОР МЕТОДОВ ДЛЯ ПОСТРОЕНИЯ МОДЕРНИЗИРОВАННОЙ СИСТЕМЫ УПРАВЛЕНИЯ И СИСТЕМАТИЗАЦИИ ТЕКСТОВОЙ ИНФОРМАЦИИ

В рамках данной главы проводится анализ основных подходов, применяемых для автоматического управления текстовыми документами, и обосновывается выбор базовых методов для построения модернизированной автоматической компьютерной системы управления и систематизации специализированной текстовой информацией.

Большое количество слов в документе является одной из проблем анализа текстов. Время поиска новых знаний резко возрастет, если каждое из слов в документе подвергать анализу. Также стоит учитывать, что не все слова в тексте несут полезную информацию. Как следствие, приведение к единой форме близких по смыслу слов, а также удаление неинформативных слов значительно сокращают время анализа текстов [54 с. 30].

Для обработки текстовой информации и применения общепринятых методов оценки результатов классификации проанализированы базовые технологии машинного обучения и лингвистические процессы естественного языка. Стоит отметить, что в рамках данного анализа сложно покрыть весь спектр методов и технологий, применяемых для автоматической обработки текстов [55 с. 262].

На основе выполненного анализа обосновывается выбор базовых методов для построения модернизированной автоматической компьютерной системы управления и систематизации текстовой информации с целью повышения их эффективности, для решения задачи построения универсального системного способа представления знаний [56 с. 328].

2.1 Основные свойства текста и методы его обработки

Системы обработки текстовой информации относятся к сложным системам, функционирование которых осуществляется в условиях недостаточных (а в ряде случаев и противоречивых) знаний о структуре документальных массивов, и затрудняется из–за действия свойств, представленных в соответствии с рисунком 2.1 [57 с. 6].

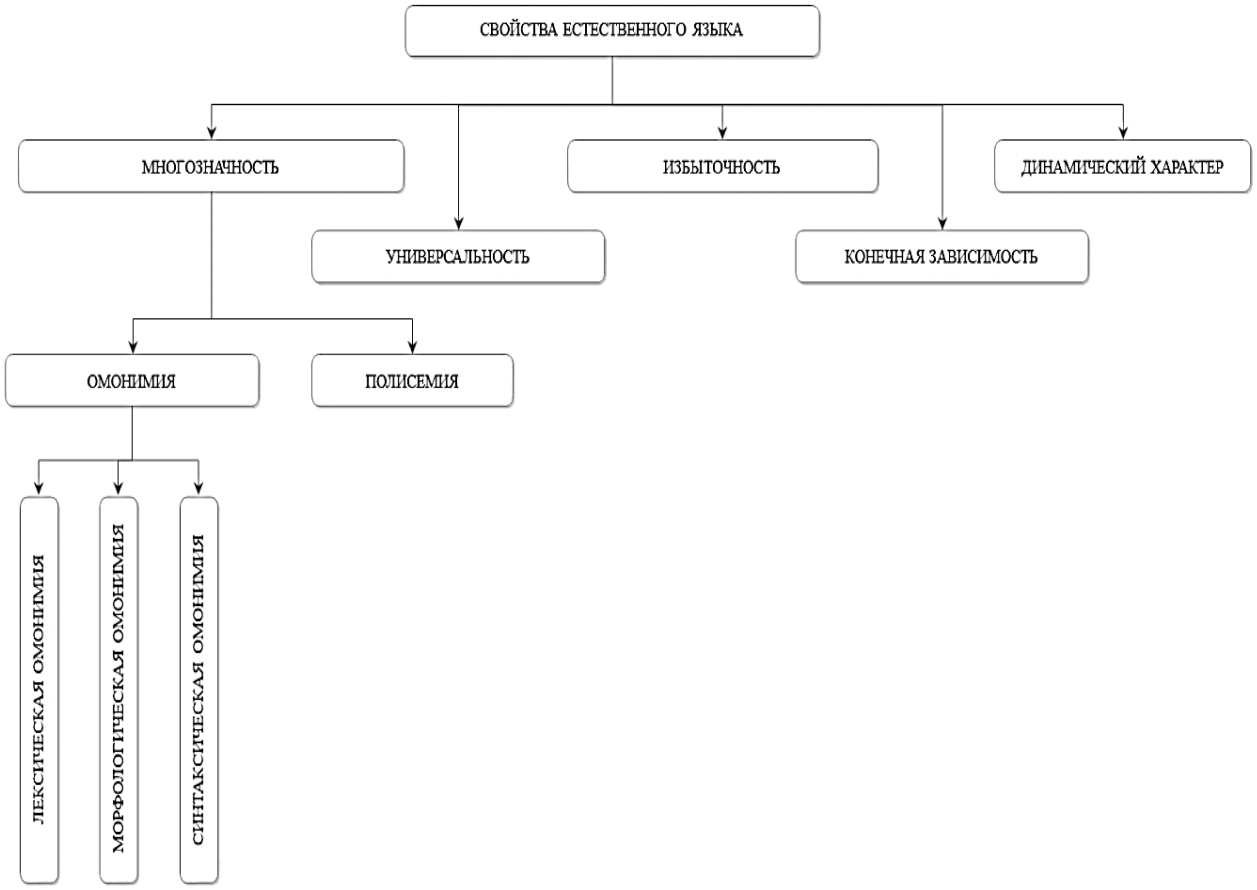

Рисунок 2.1 – Свойства русского языка

«Многозначность – наличие нескольких значений у отдельных слов и сообщений». Полисемия и омонимия, частые явления проявления многозначности [57 с. 44].

«Полисемия – использование одного слова для обозначения различных предметов и явлений». Так, например, «воинская команда призывников» – «спортивная команда» [57 с. 19].

«Омонимия – совпадение написания и/или произношения слов или иных языковых конструкций, которые имеют разные значения» [57 с. 19].

Омонимия разделяется на представленные далее типы: слова различные по значению и одинаковые по написанию – лексическая омонимия; у различных сов совпадают части словоформ – морфологическая; различные значения совпадающих в написании предложений – синтаксическая омонимия [57 с. 19].

Явление, тесно связанное с данным свойством, определяется как синонимия, другими словами, близость по значению разных выражений и слов (например, «бортовой самописец» – «черный ящик», «субмарина» – «подлодка»)».

Универсальность – способность языка передавать информацию о явлении, факте, любом событии, ситуации представляемого и реального мира. Многочисленность форм для представления грамматических конструкций и большой объем лексики (в русском языке порядка 500 000 слов) является следствием универсальности языка [57 с.70].

Зависимость интерпретации критериев от контекста и отсутствие заданной применимости слов является неопределенностью. Проявляется данное свойство, например, наличием подразумеваемых связей между словами и слов, присутствие в тексте которых не явно проявляется в таком явлении, как эллипсность [58 с. 78].

Постоянное развитие и изменение основных элементов языка (грамматики, лексики, алфавита) – изменчивость. Отсутствие строгой схемы построения высказываний является следствием данного свойства [58 с. 80].

Системный анализ используется для формализации процесса управления и систематизации текстовой информации и устранения негативного действия вышеуказанных факторов. Его применение является методологической основой исследования сложных систем и представляет мощный инструментарий для

обоснования выбора наилучших решений с точки зрения сформулированного целевого критерия.

2.1.1 Методы автоматического анализа текста

Взаимно противоречивые признаки определяют выбор формализма для представления знаний о языке [59]:

- «лингвистическая естественность»;
- «формальная мощность»;
- «вычислительная эффективность».

С одной стороны, возможность достаточно общим образом описывать феномены, относящиеся ко многим естественным языкам, то есть типологическая адекватность, а с другой стороны – удобство отображения феноменов естественного языка понимается под лингвистической естественностью. Представление в виде набора последовательно работающих процессов является следствием представление системы автоматической обработки текстов [60].

На различных уровнях осуществляется рассмотрение текста на естественном языке: «слов», «отдельных знаков», «текста в целом», «предложений».

Каждый уровень, различных разделов лингвистики является объектом рассмотрения, а именно [61]:

- «морфология»;
- «синтаксис»;
- «семантика».

«Морфология (от греческого «учение о форме») – раздел лингвистики, изучающий слова естественных языков и их значимые части – морфемы». Описание внутренней структуры и определение слова как особого языкового объекта входит в задачи морфологии. Области наиболее частого деления морфологии определяются в соответствии с двумя критериями: грамматическую семантику, изучающую свойства грамматических морфологических значений и

категорий, другими словами морфологически выражаемое словоизменение и словообразование языков мира и формальную морфологию (морфемику), в центре которой находятся понятия морфемы и слова [62 с. 11].

Совокупность правил грамматики языка, относящихся к построению единиц более протяженных, чем слово, – предложений и словосочетаний определяют синтаксис в традиционном понимании. Проблемы при изучении синтаксиса делятся на две большие группы: теоретические и описательные. В формулировке правил с наибольшей точностью и полнотой, заключается цель синтаксического описания. Построение правил, отличает неправильно построенные от правильно построенных предложений некоторого языка. Задачей теоретического синтаксиса является часть общей теории грамматики, заключающейся в выделении универсального, то есть свойственного всем языкам компонента синтаксических правил и определению пределов того разнообразия, которое проявляют языки в области синтаксиса [63].

Семантика – анализ отношения между миром и языковыми выражениями, воображаемым или реальным, а также совокупность таких отношений и само это отношение. Данное отношение заключается в том, что языковые выражения (словосочетания, слова, тексты, предложения) обозначают то, что есть в мире, а именно: качества или свойства, предметы, способы совершения действий, действия, ситуации и их последовательность и отношения [62 с. 13].

Лингвистические процессы последовательно обрабатывающие получаемую информацию являются компонентами составляющими структуру систем анализа текстов.

В системы обработки текстов входят компоненты, представленные в соответствии с рисунком 2.2 [57 с. 19].

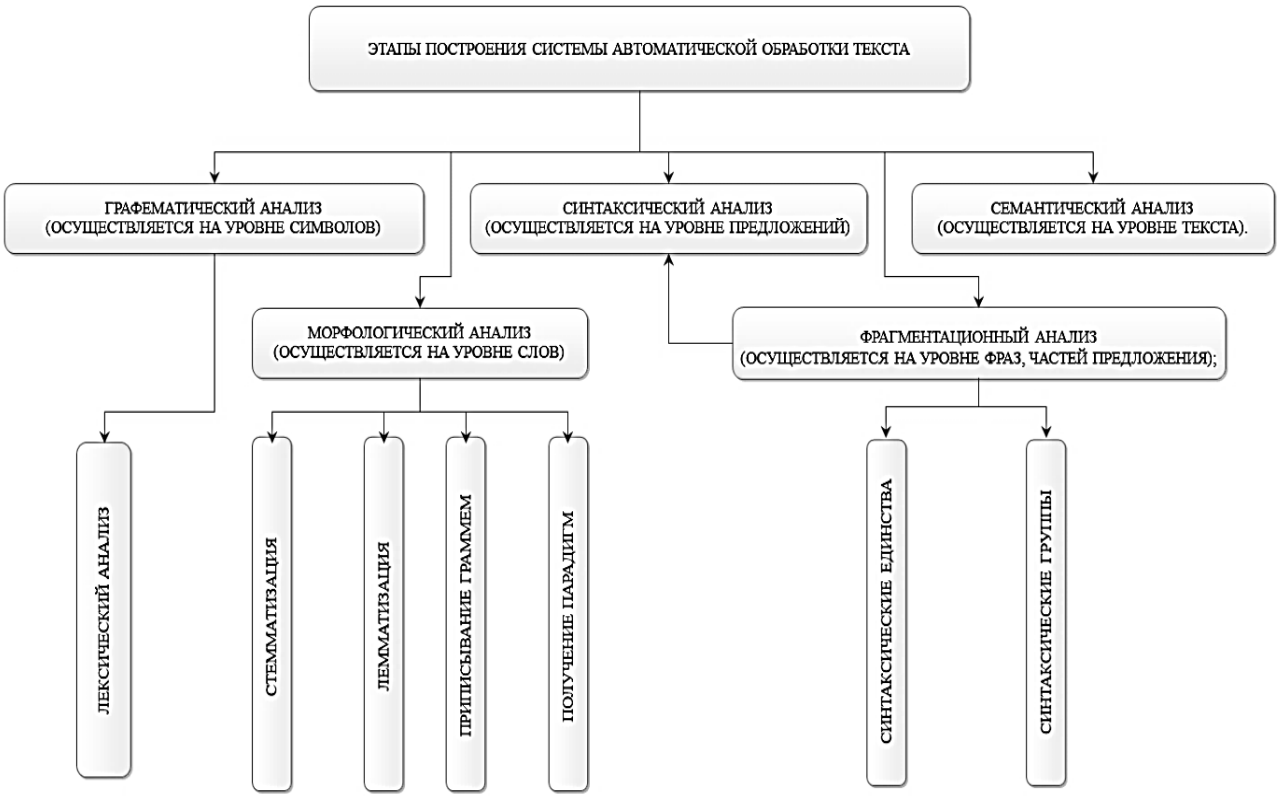

Рисунок 2.2 – Основные этапы построения систем автоматической обработки текстов.

Дискурс–анализ (прагматический анализ) является еще одним этапом анализа текста. Определение цели, которую преследует автор при изложении своих мыслей его основная задача. Но стоит учитывать, что данный этап не включен в процесс построения систем автоматической обработки текстов [62 с. 30].

Рассмотрим более подробно каждый из этапов.

Графематический анализ сводится к совмещению с морфологическим анализом и выполнению только лексического анализа. Токенизация (лексический анализ) – выделение в тексте цифровых комплексов, слов, формул, знаков препинания и т. д. Токенами или лексемами называют все выделенные элементы [58 с. 33].

Иногда осуществляют деление текста на более крупные, чем отдельные единицы, слова, иногда осуществляется путем выделения отступов, знаков

препинания, пробелов, которые служат ориентирами для разбиения абзацев и предложений.

Сегментация является подзадачей графематического анализа, которая заключается в поиске границ между словами в тексте без пробелов (например, на японском или китайском языках).

Фрагментационный анализ ставит своей целью деление предложения на неразрывные фрагменты (синтаксические единства), большие или равные словосочетания (синтаксическая группа), и установление частичной иерархии на множестве этих единств. Возможные типы фрагментов: главные предложения, придаточные предложения в составе сложного, причастные, деепричастные и другие обособленные обороты [64]. Часто этот этап совмещен с синтаксическим анализом.

Важной задачей фрагментационного анализа является установление иерархии. Любое простое предложение может быть «разорвано» «вклинивающимися» деепричастными или причастными оборотами, или придаточными предложениями, которые, также могут быть «разбиты» другими оборотами. Порой «куски» цельного высказывания находятся на значительном расстоянии друг от друга, а глубина вложения таких «клиньев» теоретически не ограничена [65].

Часто этап фрагментационного анализа совмещен с синтаксическим анализом, но так же выделяют отдельный этап, для сокращения времени работы синтаксического анализатора. В таком случае предложение по фрагментам поступает на вход синтаксическому анализу.

Синтаксический анализ осуществляет построение групп в предложении. Основная задача синтаксического анализатора – удаление значительной части «морфологического шума» и омонимичности словоформ. Важно, чтобы при синтаксическом анализе не появлялись лишние синтаксические связи, которые может допускать морфология. Это связано с морфологической неоднозначностью. Например, слово мыла не является морфологически однозначным, т. к. в зависимости от контекста может выступать в роли

существительного в единственном числе, в родительном падеже, или в роли глагола в прошедшем времени [56 с. 94].

Цель семантического анализа – построить семантический граф текста. На этапе семантического анализа, в отличие от морфологического и синтаксического, появляется формальное представление смысла текста. В сферу семантического анализа входит построение семантической интерпретации слов и конструкций и установление «содержательных» семантических отношений между элементами текста, которые уже не ограничены размером одного слова [62 с. 13].

Именно на этапе семантического анализа из всех возможных получается наиболее полное и законченное представление текста. Самой трудно решаемой задачей из пяти перечисленных в автоматической обработке текстов является семантический анализ.

На сайте NLPub[1] приведен наиболее полный список инструментов для автоматического анализа текстов.

Существует большое количество готовых синтаксических и морфологических анализаторов, как с закрытым, так и с открытым кодом: MyStem, AOT, Link Grammar и др. Также существует большое разнообразие средств разработки для создания своих приложений: Solarix[2]; библиотеки для Python (pymorphy 2, nltk); библиотеки для PHP (php Morphy) и пр.

В морфологический анализатор входит набор алгоритмов, которые занимаются сопоставлением отдельных словоформ и слов в словаре (лексиконе, если быть точным) и выяснением грамматических характеристик слов [66 с. 30].

Выделяют два типа морфологических анализаторов [57 с. 16]:

- «словарные (точный морфологический анализ)»;
- «бессловарные (приближенный морфологический анализ или аналитические методы)».

Словарные анализаторы используют методы, на словарях.

[1]URL:https://nlpub.ru/

[2]URL:http://www.solarix.ru

В словарных анализаторах преобразование слова в лемму производится с помощью специальной таблицы (словаря), которая содержит отображение множества слов на множество лемм» [67].

В качестве примера морфологического словаря, содержащего около 100 тыс. базовых словоформ русского языка с их полным морфологическим описанием, используемым при автоматическом анализе, можно привести словарь А. А. Зализняка[3]. Этот словарь является обратным – слова в нем упорядочены, начиная с последней буквы. В основу большинства современных компьютерных программ, работающих с русской морфологией, легла электронная версия этого словаря [55 с. 109].

Для определения части речи у каждого слова существуют характерные окончания служащие принципами, на которые опираются словарные анализаторы. Определить псевдооснову и основную форму слова можно отделив окончания, а также, если потребуется, суффиксы в словах. Другими словами, для каждой части речи конкретным словам ставятся в соответствие векторы окончаний. Число словоформ данной части речи равно длине вектора окончаний [55 с. 109].

Если в словаре нет той или иной морфологической информации о словоформе подвергшейся анализу, то ее получение становится невозможным, что является главным недостатком словарных анализаторов. Учитывая, что порядка 80% словарного запаса составляет неизменяемое лексическое ядро естественного языка, использование только словарных анализаторов не решает проблему определения в полном объеме всех возможных слов [55 с. 109].

Бессловарные анализаторы используют аналитические методы и содержат набор правил морфологических преобразований. Для русского языка это в основном таблицы суффиксов и условий их отсечения, с помощью которых данное слово преобразуется в некоторую нормальную форму [66 с. 41].

[3]URL:http://zaliznyak–dict.narod.ru

Во времена существенного ограничения оперативной памяти появились бессловарные морфологические словари. Словарные морфологии получили свое распространение с увеличением количества мегабайт или даже десятков мегабайт оперативной памяти, наличие которых не составляет проблемы в настоящее время [55 с. 109].

В качестве нормальной формы взяв неизменяемую псевдооснову, называемую стемом, а также отбросив всю морфологическую информацию можно выделить систему на основе стемминга. Лексическая морфология проводится аналогичным образом с анализом в подобных системах [66 с. 45].

За счет уменьшения объема выдаваемой информации и упрощения алгоритма скорость анализа существенно, до нескольких раз, возрастает, а объем хранимых баз сократится при использовании лишь массива парадигм, что является неотъемлемым достоинством стемминга на основе морфологии [58 с. 8].

Получение неограниченного объема морфологической базы непосредственно настраиваемой на имеющийся текст, при отсутствии словаря основ является главным достоинством морфологии на основе стемминга. Нефиксированная лексика, используемая для сознания информационно–поисковых систем удобна в использовании. Морфология никогда не сообщает нам, что такого слова нет в словаре, так как некоторый набор стемов мы получаем при индексировании текстов, который заносим в индекс [58 с. 8].

Невысокая точность метода, а также невозможность морфологического синтеза на базе без основ является недостатком данного подхода.

Следует заметить, что грань между стемминговой морфологией, базирующейся на неизменяемой псевдооснове и лексической морфологии, дающей полный набор морфологических параметров и оперирующей с нормальными формами слова, довольно тонка. С одной стороны, лексическая морфология использует неизменяемую основу, то есть стем. С другой стороны, при хранении полного набора лексической информации стемминг отличается от лемматизации лишь выдаваемой строкой нормальной формы. Система морфологического анализа MyStem хотя и называется стеммером, точнее

– парсером, компании Яндекс[4], предоставляет полный набор лексической информации о слове. Морфологический словарь «Диалинг»[5] аналогичный по выдаваемым характеристикам и объемам ни в коем случае не заявляется, как стеммер, а является полноценным лемматизатором [58 с. 40].

Аналитические методы, не применяющие словари, являются одной из современных вариаций реализации бессловарной морфологии. Определение грамматических признаков словоформы и частей речи затрудняют решения задач морфологического анализа. Стоит учитывать, что для создания процедур работы со словарями естественных языков и задач индексации текстовых массивов аналитические алгоритмы оказываются эффективными [62 с. 17].

Аналитические методы, не применяющие словари, являются некоторым вариантом современного воплощения бессловарной морфологии. Грамматические признаки словоформ и сложность определения частей речи, входящие в задачи морфологического анализа, не решаются аналитическими методами. Стоит учитывать, что аналитические алгоритмы показывают свою эффективность в задачах создания процедур работы со словарями естественных языков, и при индексации текстовых массивов [62 с. 17].

Для повышения эффективности решения проблемы аналитического выделения основ на практике применяется смешанный подход, то есть наряду с правилами преобразования аффиксов рассматриваются таблицы, например, неправильных глаголов или исключений при образовании форм множественного числа имен существительных [68].

Отсечение аффиксов используется в алгоритмах аналитического выделения основ, разрабатывающихся в соответствии с грамматикой каждого конкретного языка. При создании процедур автоматической обработки текстов на естественном языке разработчики информационных систем склонны рассматривать это обстоятельство как существенный сдерживающий момент [62 с. 24].

[4] URL:http://company.yandex.ru/technology/mystem

[5]URL: http://www.aot.ru/docs/sokirko/sokirko-candid-1.html

Такие алгоритмы используют частоты *N*–грамм, скрытые Марковские модели и нейронные сети. Универсальные методы выделения основ, не опирающиеся на грамматику естественного языка, требуют обработки больших объемов текстовой информации для обучения системы или в процессе работы самой информационной системы, систематизирующей информацию. Это обстоятельство не позволяет достигать высоких скоростей и качества обработки текстов, что делает их плохо пригодными к использованию в реальных системах [69, 70].

Способность предоставлять результаты для любых слов, попадающихся в тексте, является существенным достоинством бессловарных морфологий, что достаточно удобно при анализе информации из незнакомой предметной области или содержащих много редко употребляемых или нелитературных слов. Однако в таких задачах, как диалоговые системы и машинный перевод, где отдают предпочтение точности анализа над его полнотой, бессловарные морфологии показывают свою недостаточную эффективность, так как корректность их работы, находится на уровне 90–95%, что привело к отказу от них. На практике существует достаточное количество задач, в которых вполне достаточно приблизительных знаний о связях между словами, решение которых предоставляется статистическим методам. Это задачи рубрикации, информационного поиска, частично – задачи реферирования, ряд других задач [71].

Методы бессловарных морфологий активно используются в словарных морфологиях для предсказания нормальной формы и набора параметров слов, которые отсутствуют в морфологическом словаре. Для этого необходимо проанализировать постфиксы слова и попытаться образовать нормальную форму исходя из полученного префикса и парадигмы, приписываемой постфиксу. Для этого по найденным постфиксам определяются постфиксы нормальной формы, которые присоединяются к полученным префиксам, и наборы морфологических параметров.

Существенным недостатком является большое количество предсказанных вариантов [58 с. 33].

Фильтрация используется для существенного сокращения количества таких вариантов при наличии достаточно большого морфологического словаря можно сделать вывод о том, что в нем находятся все союзы, предлоги, местоимения, и другие части речи. Используя статистику также можно произвести отсев достаточно большого количества парадигм, в которые входит всего по нескольку слов. Эти парадигмы в большинстве своем являются закрытыми, то есть добавление новых слов в них уже невозможно. В связи с этим можно отсечь подобные парадигмы, запретив выдвижение гипотез на их основе. Для этого достаточно подсчитать количество слов, относящихся к каждой из гипотез, и выдвигать гипотезы только на основе парадигмы, к которым принадлежит количество слов, большее заданного порога [58 с. 13].

Слова, принадлежащие одной парадигме, совпадающие не только изменяемыми частями, но и последними несколькими символами неизменяемой являются еще одним способом отсеивания. Для увеличения словаря применяются методы бессловарных морфологий.

Большинство современных алгоритмов машинного обучения ориентированы на признаковое описание объектов, поэтому все документы обычно переводят в вещественное пространство признаков. Для этого используют идею о том, что за принадлежность документа к некоторому классу отвечают слова, а тексты из одного класса будут использовать много схожих слов [58 с. 14].

Статистическая информация о словах позволяет использовать преимущественно известные способы, осуществляющие преобразование текста в пространство признаков. При использовании которых, каждый объект преобразуется в вектор, длина которого приравнивается количеству слов, входящих в текст выборки.

2.2 Векторное представление документов

Текстовые документы в непосредственном виде не подходят для интерпретации классификатором или алгоритмом построения классификатора. Поэтому необходимо применение процедуры индексации, которая переводит текст в удобное для классификатора представление.

В традициях информационного поиска каждый документ, как правило, представляется в виде вектора состоящего из *n* взвешенных терминов в соответствии с рисунком 2.3.

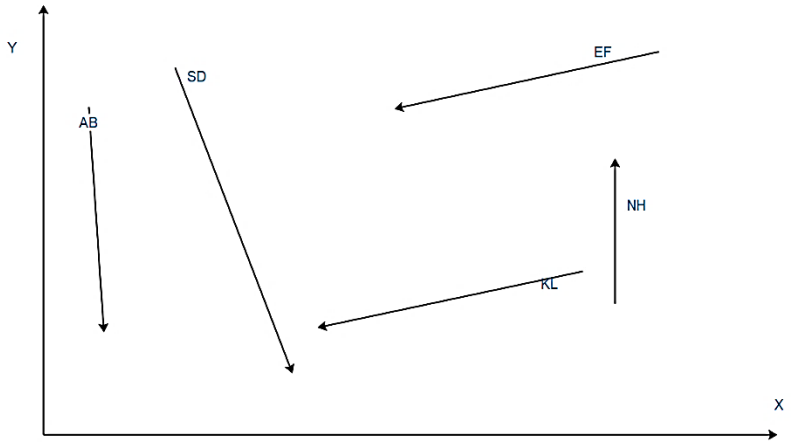

Рисунок 2.3 – Примеры векторов

Решать задачи тематического моделирования можно используя векторную модель для представления текста (*Vector Space Model, VSM*) [56 с. 328]. Изначально данная модель использовалась в 1996 – 1997 годах в задачах определения тем (*Topic Detection and Tracking*) путем получения событий из потока информации. Представление корпуса текста в данном случае происходит с помощью векторов из одного общего для всей коллекции векторного пространства, в котором каждому слову сопоставляется вес в соответствии с выбранной весовой функцией. Указание, каким именно образом будет определяться вес слова в документе, является необходимостью для полного определения векторной модели. Для чего применяются различные методы:

оформление слова, место появления слова, статистический подход (логарифм вхождения слова в текст, *TF–IDF*, Булевский вес и прочие) и другие. Представить, таким образом текст можно, например, решая задачу подобия документов или определяя расстояние между точками – чем ближе расположены точки, тем более схожи рассматриваемые тексты [72].

На статистической информации о словах основаны наиболее известные способы, позволяющие воплотить перевод текста в пространство признаков. Использование которых, подразумевает перевод каждого объекта в вектор, длина которого равна количеству используемых слов во всех текстах выборки.

BagofWords&TFIDF - популярный способ перевода текста в векторное представление в котором документы представляются в виде матрицы $T = (t)_{d,w}$, но элемент $(t)_{d,w}$ функции $TF - IDF(w;\ d;\ D)$ слова $w\ \in\ W_d$ в документе $d\ \in\ D$ [73].

Для оценки важности слова в контексте документа, являющегося частью корпуса или коллекции документов, используется статистическая мера *TF–IDF* [74]. В которой «вес некоторого слова пропорционален количеству употребления этого слова в документе и обратно пропорционален частоте употребления слова в других документах коллекции». Вычисляется по формуле [75]:

$$TF - IDF(w, d, D)\ =\ TF(w, d)\ \times IDF(w, D). \qquad (2.1)$$

TF (*term frequency* – частота терма) – «отношение числа вхождения некоторого терма к общему количеству термов документа. Другими словами, оценивается важность терма w в пределах отдельного документа d, частота слова, оценивает важность слова w_i в пределах документа» [76].

$$TF(w, d)\ = \frac{n_i}{\sum_k n_k}\ , \qquad (2.2)$$

где

n_i – число вхождений слова i в документ.

$\sum_k n_k$ – общее число слов в данном документе.

IDF (*inverse document frequency* – обратная частота документа) – «инверсия частоты, с которой некоторое слово встречается в документах коллекции». Учёт *IDF* уменьшает вес широкоупотребительных слов. Для каждого уникального слова в пределах конкретной коллекции документов существует только одно значение *IDF* [77].

$$IDF(w, D) = log \frac{|D|}{|(d_i \supset w_i)|}, \quad (2.3)$$

где

$|D|$ – количество документов в корпусе.

$|(d_i \supset w_i)|$ – количество документов, в которых встречается слово w_i.

Привлечение дополнительной информации также используется в некоторых случаях для вычисления веса слова в тексте. Например, можно словам, встреченным в заголовке, присваивать больший вес и учитывать информацию о структуре текста [78].

Существуют также другие вариации формулы *TF–IDF*, которые предоставляют близкие по качеству, но более низкие результаты, опишем некоторые из них.

Модель мешок слов (*BagofWords*) используется для перевода текста в векторное представление. Основное предположение данного метода заключается в отсутствии важности порядка слов в документе, а коллекция текстовых документов рассматривается как простая выборка пар «документ – слово» $(d; w)$, где $d \in D, w \in W_d$ [79].

Некоторая последовательность слов также несет полезную информацию в тексте. Например, фразеологизмы – устойчивые сочетание слов или использования речевого оборота «Как рыба в воде» означает очень хорошо в чем–либо разбираться или чувствовать себя уверенно. Смысл данного выражения будет потерян, если каждое его слово учитывать по отдельности [80].

N–граммы - это последовательности из *N* слов, которые позволяют учесть данные особенности языка при преобразовании текстов в векторное пространство

признаков. В задаче классификации текстов *N*–граммы являются индикаторами того, что данные *N* слов встретились рядом. Например, для текста «мама мыла раму» получаем биграммы «мама мыла» и «мыла раму» [81, 82].

Метод *BagofNgrams & TF–IDF* подобен методу *BagofWords& TF–IDF*, но различия состоят в том, что вектор признаков для каждого документа помимо *TF–IDF* слов содержит *TF–IDF* всех последовательностей из *n* слов [83].

Привязка к частотной характеристике (обратной частоте документа), которая заметно занижает вес распространённых слов, несущих не высокую смысловую нагрузку можно отнести к преимуществам метода *TF–IDF*.

Существенное занижение веса документов, документов большой длины, включающих схожие определения, которые определенно будут проигрывать по *TF* коэффициенту, и присвоение слишком большого веса «коротким» документам, по этой же причине можно отнести к недостаткам метода. Сокращения числа используемых атрибутов, путем выделения наиболее значимых, используется для борьбы с высокой размерностью.

2.3 Методы машинного обучения

2.3.1 Алгоритм *k*–ближайших соседей

«Метод *k*–ближайших соседей (*k–nearest neighbours, k–NN*), в отличие от других, не требует фазы обучения». Данный алгоритм, имеет достаточно большое количество трансформаций [84].

Основной алгоритм использует гипотезу компактности векторного пространства, которая заключается в том, что документы одного класса образуют в пространстве терминов компактную область, причём области разных классов не пересекаются. Как следствие появляется вероятность того, что тестовый документ будет иметь метку класса аналогичную окружающим его документам из обучающего множества. Алгоритм *k*–ближайшего соседа распределяет новый

тестовый документ к преобладающему классу его k соседей в соответствии с рисунком 2.4 [85].

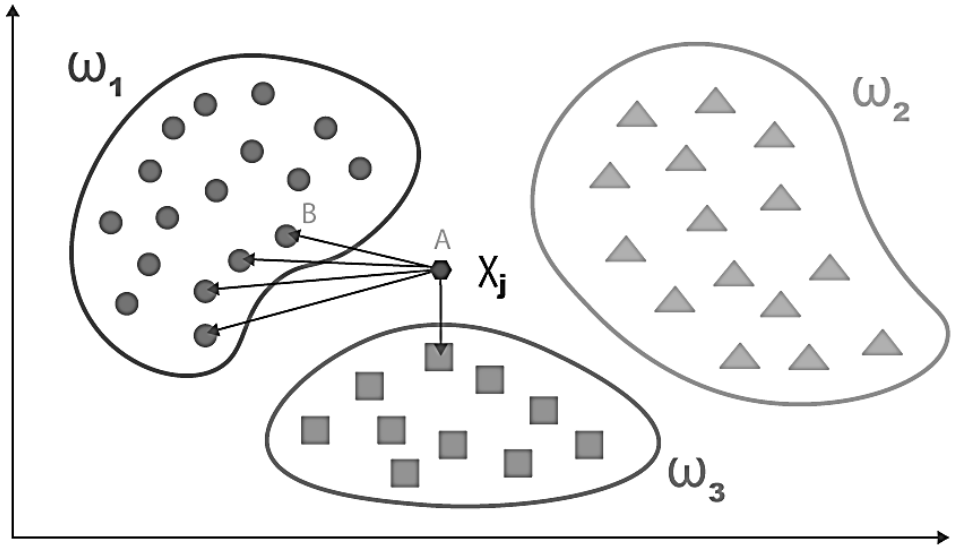

Рисунок 2.4 – Метод *k*–ближайших соседей

Документ сопоставляется со всеми имевшимися документами из обучающей выборки для определения рубрики, релевантной документу d. Из обучающей выборки для каждого документа d, находится расстояние – косинус угла между векторами признаков [86]:

$$\rho(d, e) = cos(d, e). \quad (2.4)$$

После этого из обучающей выборки выбираются k документов, ближайших к d (*k*– параметр). Релевантность для каждой рубрики вычисляется по формуле:

$$s(c_j, d) = \sum_{e \in \{k \text{ ближайших соседей}\} \wedge c_j \in Rub(e)} cos(d, e). \quad (2.5)$$

Соответствующими документу считаются рубрики, у которых релевантность выше заданного порога. Параметр k обычно присваивается из интервала от 1 до 100.

Данный алгоритм имеет отношение к группе алгоритмов, основанных на представлении кластеров с помощью эталонов (типичных, наиболее характерных представителей). Подробно метод *K–NN* изложен в [87 с. 43].

Преимущества и недостатки метода представлены в таблице 2.1.

Таблица 2.1 – Особенности работы алгоритма *K–NN*

Достоинства	Недостатки
1	2
Поддерживает возможность обновлять обучающую выборку без переобучения классификатора, так как нет необходимости в построении классифицирующей функции.	Подборка текстовых данных, применяемых для алгоритма, должна быть показательной (репрезентативным).
К аномальным выбросам в исходных данных, алгоритм устойчив.	Результаты классификации значительно зависят от выбранной метрики.
Программная реализация алгоритма относительно проста.	Необходимость полного перебора обучающей выборки влияет на значительное увеличение длительности работы.
Интерпретации, легко поддаются результаты работы алгоритма.	Не подходит для решения задач большой размерности, по количеству классов и документов.
Хорошо справляется с линейно неразделимыми выборками.	–

Вычислительная сложность алгоритма определяется так.

Обучение: $O(|\Omega|)$.

Тестирование: $O(|C|)$.

Данный алгоритм иногда используется в задачах регрессии, так как считается одним из простейших алгоритмов классификации. На практических задачах он нередко демонстрирует свою неэффективность. Скорость

классификации является еще одним недостатком помимо точности классификации для алгоритма *K–NN* [87 с. 43].

2.3.2 Алгоритм наивного Байесового классификатора

В некоторых случаях для решения задач классификации возникает потребность в использовании нескольких независимых переменных. Справиться с данной задачей классификации, позволяет выполнение алгоритма Naive Bayes, который для расчета вероятности применяет формулу Байеса. Предположение о том, что все рассматриваемые переменные независимы друг от друга, определило название naive (наивный). На практике это не всегда так, однако, данный алгоритм находит свое применение [88].

«Наивный байесовский классификатор (*Naive Bayes Classifier*, *NBC*) является практичным и известным вероятностным классификатором, нашедшим свое применение во многих приложениях» [5 с. 38].

Формула Байеса для расчета вероятности входит в алгоритм выполняющий классификацию [89].

$$P(c|d) = \frac{P(d|c)P(c)}{P(d)}, \quad (2.6)$$

где

$P(c|d)$ – вероятность, что документ d принадлежит классу c, именно её надо рассчитать;

$P(d|c)$ – вероятность встретить документ d среди всех документов класса c;

$P(c)$ – безусловная вероятность встретить документ класса c в корпусе документов;

$P(d)$ – безусловная вероятность встретить документ d в корпусе документов.

Переставить местами следствие и причину позволяет Теорема Байеса. Данная теорема позволяет рассчитать вероятность того, что именно эта причина

привела к наблюдаемому событию, с условием того, что будет известно, с какой вероятностью причина приведет к некоему событию [90].

Таким образом, для реализации Байесовского классификатора необходима обучающая выборка, в которой проставлены соответствия между текстовыми документами и их классами. Затем нужно собрать следующую статистику из выборки, которая будет использоваться на этапе классификации:

- относительные частоты классов в корпусе документов, то есть, частота встречаемости документа, принадлежащего тому или иному классу;
- суммарное количество слов в документах каждого класса;
- относительные частоты слов в пределах каждого класса;
- размер словаря выборки, количество уникальных слов в выборке [91].

Совокупность этих данных условно назовем моделью классификатора.

Используя формулы Байеса для оценки достоверности правил возникает проблема, состоящая в том, что в обучающей выборке может не оказаться ни одного объекта, имеющего значение c_d^h переменной x_h и относящегося к классу c_r. В данной ситуации соответствующая вероятность будет равна 0, и как следствие, вероятность такого правила равна 0. Чтобы уклонится от этого, необходимо к каждой вероятности присоединить некоторое значение, не равное нулю, в результате чего появится методика, называемая оценочной функцией Лапласа [5 с. 38].

Пропущенные значения в данном методе не создают никакой проблемы, что является одним из преимуществ. При подсчете вероятности они просто пропускаются для всех правил, что в свою очередь не влияет на соотношение вероятностей. Достоинства и недостатки метода представлены в таблице 2.2.

Таблица 2.2 – Особенности работы метода *NBC*

Достоинства	Недостатки
1	2
Быстрое и легкое выполнение многоклассовой классификации.	Если в тестовом наборе данных присутствует некоторое значение категорийного признака,

Продолжение таблицы 2.2

Достоинства	Недостатки
1	2
	которое не встречалось в обучающем наборе данных, тогда модель присвоит нулевую вероятность этому значению и не сможет сделать прогноз.
При условии, если допущение о независимости выполняется, *NBC* превосходит другие алгоритмы, такие как логистическая регрессия (*logistic regression*), и при этом требует меньший объем обучающих данных.	Значения спрогнозированных вероятностей не всегда являются достаточно точными, не следует слишком полагаться на результаты, полученные методом *NBC*.
NBC лучше работает с категорийными признаками, чем с непрерывными. Для непрерывных признаков предполагается нормальное распределение, что является достаточно сильным допущением.	Допущение о независимости признаков. В реальности наборы полностью независимых признаков встречаются крайне редко.

Вычислительная сложность алгоритма определяется так:

Обучение: линейная сложность относительно размера коллекции документов $O(|\Omega|)$.

Тестирование: линейная сложность относительно числа категории $O(|C|)$.

Несмотря на то, что предположение об условной независимости, как правило, не соответствует действительности для появления слова в документах, наивный байесовский классификатор показывает достаточно хорошие результаты при несоблюдении условия статистической независимости [92].

2.3.3 Алгоритм опорных векторов *SVM*

Алгоритм, разработанный В. Вапником, на основе принципа структурной минимизации риска, другими словами одновременного контроля количества

ошибок классификации на множестве, для обучения и «степени обобщения» обнаруженных зависимостей называется Метод опорных векторов (*Support Vector Machines, SVM*) [93 с. 15].

Основная идея метода опорных векторов заключается в переводе исходных векторов в пространство более высокой размерности и осуществление поиска разделяющей гиперплоскости с максимальным зазором в данном пространстве. Разделение на классы осуществляется при помощи двух параллельных гиперплоскостей, строящихся по обеим сторонам от гиперплоскости. За разделяющую гиперплоскость принимается гиперплоскость, максимизирующая расстояние до двух параллельных гиперплоскостей. Работа алгоритма проходит в предположении, что чем больше расстояние или разница между данными параллельными гиперплоскостями, тем больше будет снижена средняя ошибка классификатора [94].

Предполагается, что существует коллекция – это множество векторов $\{x_1, \dots, x_n\} \in R^N$, чисел $\{y_1, \dots, y_n\} \in \{-1,1\}$. Число y_i равно *1* в случае принадлежности соответствующего вектора x_i категории *c*, и *–1* – в противном случае. Простейший способ решения задачи классификации заключается в использовании линейного классификатора. В этом случае ищется прямая, другими словами гиперплоскость в *N*–мерном пространстве, разделяющая все точки по принадлежащим им классам. При условии возможности определения такай прямой, задача классификации сводится к определению взаимного расположения точки и прямой: если новая точка лежит с одной стороны прямой, гиперплоскости, то она принадлежит классу *c*, если с другой стороны – классу $\bar{c}$ в соответствии с рисунком 2.5 [5 с. 223].

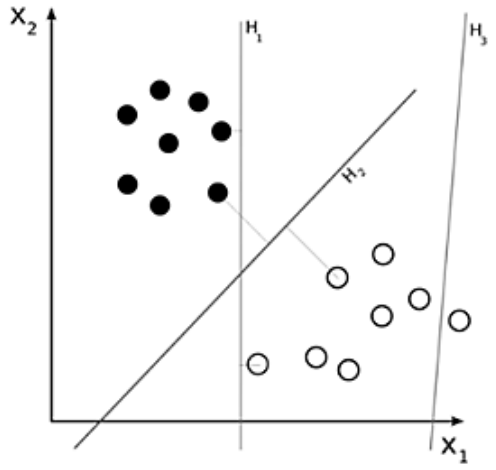

Рисунок 2.5 – Выбор разделяющей гиперплоскости

В том случае, когда отрицательные и положительные экземпляры обучающей выборки линейно не разделимы, этот метод также применим. Скалярное произведение функции – ядра из теории Гильберта–Шмидта, которые используют вместо поиска целевой зависимости. Также стоит учитывать возможность добавления в Лагранжиан дополнительных «ослабляющих» неизвестных. Использование данного подхода для категоризации текстов было предложено в [95 с. 55].

Отсутствие необходимости отбора параметров, при условии наличия в исходном пространстве достаточно высокой размерности, предполагает, что выборка окажется линейно разделимой, что является достоинством механизма опорных векторов. Более подробно достоинства и недостатки метода *SVM* для классификации текстов, представлены в таблице 2.3 [93 с. 255].

Таблица 2.3 – Особенности работы метода *SVM*

Достоинства	Недостатки
1	2
Реализация метода превосходит другие методы на тестовых массивах документов.	Малое число параметров для настройки, а именно, после того как фиксируется функция ядра, единственным параметром, который варьируется, остается коэффициент ошибки.

Продолжение таблицы 2.3

Достоинства	Недостатки
1	2
Эмуляция различных подходов становится возможной при выборе ядра, задаваемого различными функциями.	Нет четких критериев выбора функции ядра, есть лишь рекомендации, позволяющие сделать предположение об эффективности использования той или иной функции.
Итоговое правило выбирается с помощью оптимизации некоторой целевой функции, а не путем использования некоторых эвристик.	Система классификации обладает достаточно медленным обучением.
Нет ограничений по количеству плоскостей для работоспособности алгоритма.	Количество векторов, с которыми способен работать алгоритм, достаточно невелико.
–	Скорость обучения одна из самых низких.
–	Неустойчивость по отношению к выбросам в исходных данных.

Вычислительная сложность определяется так.

Обучение: $O(|C||\Omega|^2)$.

Тестирование: $O(|C|)$.

Метод опорных векторов создает гиперплоскость или набор гиперплоскостей в многомерном или бесконечномерном пространстве, которые могут быть использованы для решения задач классификации, регрессии и других близких задач.

2.3.4 Алгоритм деревья принятия решений

Дерево принятия решений представляет собой простой классификатор и широко используется в задачах классификации текстов.

Для задач классификации используют метод «Деревья решений» (*Decision tree*) предоставляющий возможность предсказывать принадлежность объектов или наблюдений к определенному классу категориальных зависимостей переменной в соответствии со значениями нескольких или одной предикторных переменных» [96].

Алгоритм *Decision tree*, подобен его прототипу из живой природы, а именно создается алгоритм, опирающийся на листья и ветви. Ветви или ребра графа хранят в себе значения атрибутов, от которых зависит целевая функция; значения целевой функции, записываются на листьях. Используют также и другие узлы – родительские и потомки, по которым происходит разветвление в соответствии с рисунком 2.6 [97].

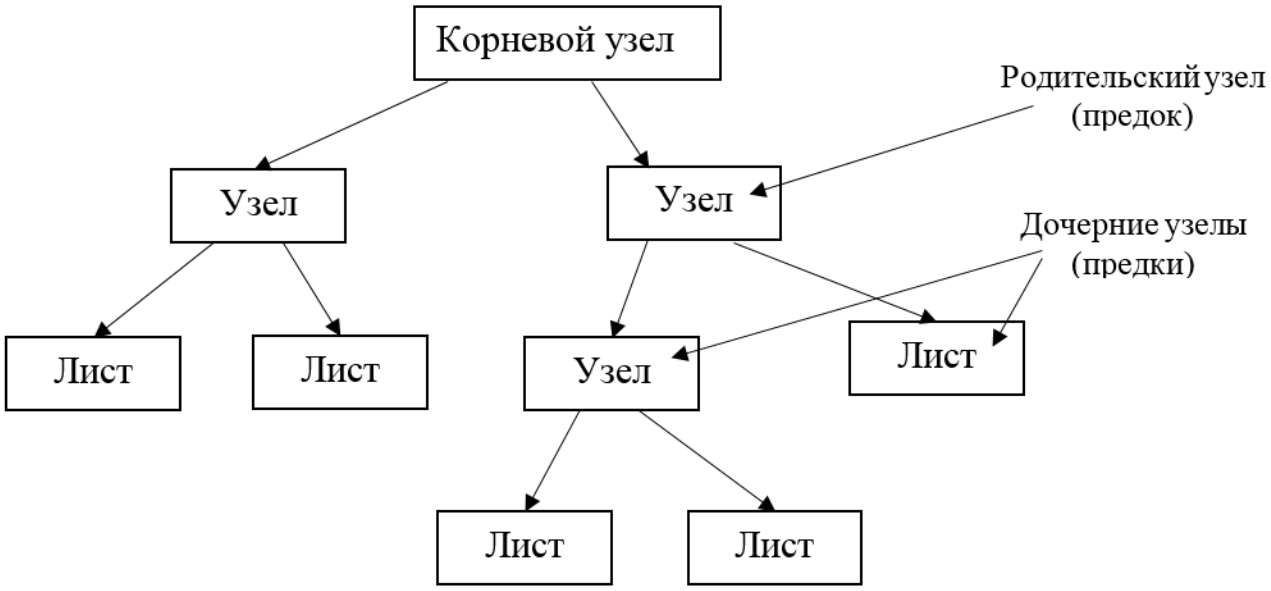

Рисунок 2.6 – Пример дерева решений

Обычно после построения «точного» дерева решений к полученному дереву применяются различные процедуры усечения и преобразования дерева для того, чтобы обеспечить баланс между сложностью дерева (количеством узлов) и качеством обучения.

Существует целый ряд стандартных пакетов для построения дерева принятия решения на основе обучающей выборки. Большинство приложений

этого метода для классификации текста обычно используют ту или иную подобную библиотеку. Среди наиболее популярных, можно назвать ID 3, C 4.5 и C 5 [98, 99].

Одним из возможных алгоритмов построения дерева принятия решений для категории c_i с помощью обучающего множества может служить стратегия «разделяй и властвуй». На каждом шаге алгоритма проверяется:

- принадлежность обучающих экземпляров к данной категории, либо c_i, либо $\bar{c}_i$;
- обучающее множество разбивается на два класса, если результат равен $\bar{c}_i$ выбирается термин t_k, вес которого постоянен, и равен либо 0 либо 1. Полученные классы относятся к разным поддеревьям.

Данный процесс продолжается до тех пор, пока все обучающие документы не будут отнесены к определенной категории, значение которой присваивается данному листу. Разбиение проходит в процессе определения подходящего термина t_k, что является ключевым моментом в данном процессе. Используя критерий энтропии, или значения прироста информации, осуществляется выбор такого терма. Стоит учитывать, что «разросшееся» дерево подвержено переобучению, так как определенные ветви могут быть слишком восприимчивы к обучающим данным [100].

Алгоритм построения дерева, и его усечения включают в себя методы на основе деревьев принятия решений, другими словами отсечение чрезмерно специфичных ветвей, для более корректной классификации тестовых документов. На основе значений переменных пространства признаков разбивают данные, на группы, используя *Decision trees*, вследствие чего, появляется иерархия операторов «ЕСЛИ–ТО», классифицирующих данные.

Принимая решения, алгоритм определяет, к какой категории относится данный документ, для этого необходимо ответить на вопросы, находящиеся в узлах данного дерева, начиная с его корня. Например, вопрос может выглядеть так: «значение переменной x_i больше порога b_t?». Если ответ, отрицательный

осуществляется переход к левому узлу, а если положительный – к правому узлу данного дерева. Далее обрабатывается вопрос, связанный с соответствующим узлом.

Разработан ряд алгоритмов для автоматического построения деревьев принятия решений с помощью обучения на примерах. Использование метода дерево принятия решений для классификации текстов обладает как достоинствами, так и недостатками, представленными в таблице 2.4 [101 с. 120].

Таблица 2.4 – Особенности работы метода дерево принятия решений

Достоинства	Недостатки
1	2
Доступен в интерпретации и понимании. После краткого объяснения люди смогут интерпретировать результаты модели дерева принятия решений.	Сложности при построении оптимального дерева принятия решений является *NP*–полной.
Способен работать как с интервальными, так и с категориальными переменными.	При изучении метода дерева принятия решений, появляется возможность создания слишком сложной конструкции, которая недостаточно точно представляет данные, в результате чего появляется проблема, называемая «чрезмерной подгонки».
Булевая логика, при возникновении определенной ситуации, наблюдаемой в модели, объясняет использование модели «белого ящика».	Описание концептов сложным путем усложняет понимание модели.
Статистические тесты способствуют оценке модели.	Больший информационный вес присваивается тем атрибутам, которые имеют большее количество уровней, для данных, включающих категориальные переменные с большим набором уровней.
Метод хорошо работает даже при условии нарушения первоначальных	–

Продолжение таблицы 2.4

Достоинства	Недостатки
1	2
предположений, включенных в модель.	
Предоставляет возможность работы с большими объемами информации без специальных подготовительных процедур и не нуждается в специальном оборудовании для работы с большими базами данных.	–

Вычислительную сложность метода можно определить так.

Обучение: $0(|\Omega| \log|\Omega|)$.

Алгоритм способствует принятию правильного решения, позволяющего систематизировать и классифицировать информацию по определенному вопросу, то есть спрогнозировать исход. Выбор основных вопросов, составляющих ключевые моменты, является важной задачей. Для облегчения поиска, а также качественного и быстрого построения дерева принятия решений существует множество моделей и компьютерных программ.

2.4 Методы усиления простых классификаторов

Методы, основанные на комбинировании «слабых» примитивных классификаторах в один, являются усилением простых классификаторов.

Повысить точность классификации для решения практических задач, согласно литературным источникам, позволит использование комбинаций классификаторов. Эмпирические и теоретические результаты демонстрируют наибольшую эффективность для комбинаций классификаторов, в случаях, когда классификаторы представляются независимыми. Обучение отдельных членов ансамбля на различающихся подмножествах признаков является наиболее эффективным методом построения независимого классификатора. Как следствие, построение ансамбля классификаторов основанных на декомпозиции исходного

набора признаков, рассматривающих объекты данных, чаще всего имеет свои преимущества [102].

Существует 18 типов алгоритмов, в которые входят: бэггинг; голосование методом Борда (*Borda count*); бустинг; нахождение «средней» и «серединной» метки класса; стекинг; адаптивное взвешивание «мнений» более точных классификаторов; смесь локальных экспертов (*Mixture of Local Experts, MLE*); ранжирование с последующей логистической регрессией на множестве рангов и другие [103].

Среди разнообразия ансамблевых методов классификации рассмотрим три типа более подробно:

- бэггинг (*bagging, bootstrap aggregating*);
- бустинг (*boosting*);
- стекинг (*stacking*) [104].

Идея бэггинга заключается в том, что существует возможность создания множества случайных выборок, состоящих из исходного простого выбора с замещением, при отсутствии большой обучающей выборки. Хотя элементы в выборках имеют возможность дублироваться или пересекаться, при решении реальных задач все же результаты объединения многих выборок оказываются точнее, чем использование одной начальной выборки. Метод имеет данное название, так как объединяет итоги предсказания разнообразных классификаторов, обученных на случайных подмножествах [105].

Бэггинг становится полезным, только при условии различных классификаторов и нестабильностей, когда следствием небольшого изменения в начальной выборке становится существенные изменения в классификации [106].

Бустинг (*boosting*) включает процедуру последовательного создания композиции алгоритмов машинного обучения, при условии, что каждый последующий алгоритм стремится компенсировать неточности композиции всех предыдущих алгоритмов. Изначально понятие бустинга появилось в работах по почти корректному обучению в связи с вопросом: возможно ли, имея множество

плохих, незначительно отличающихся от случайных, алгоритмов обучения, получить хороший алгоритм.

Стекинг (*stacking*) или просто стековое обобщение (*stacked generalization*) является еще одним средством объединения классификаторов, которое вводит понятие метаалгоритма обучения. В отличие от бустинга и бэггинга, классификаторы разной природы используются при стекинге. Идея стекинга заключается в следующем [107]:

- создать два непересекающихся подмножества из обучающей выборки;
- обучить на первом подмножестве несколько базовых классификаторов;
- на втором подмножестве протестировать базовые классификаторы;
- обучить мета алгоритм используя предсказания из предыдущего пункта как входные данные, а истинные классы объектов как выход.

В основе такого рода систем лежит идея обучения нескольких базовых, основных классификаторов на одной и той же обучающей выборке, и комбинации их предсказаний для новых тестируемых объектов [108].

По мере развития теории машинного обучения и накопления практического опыта применения различных алгоритмов стало понятно, что не существует идеального метода классификации, который был бы: лучше всех остальных при всех размерах и обучающей выборки; при любом проценте шума в данных; при любой сложности границы разделения объектов на классы и так далее. Поэтому в настоящее время активно развиваются ансамблевые методы классификации, объединяющие в одной модели множество разных классификаторов, обученных на разных выборках данных [109].

2.5 Оценка качества классификации

При решении задачи классификации текстов методами машинного обучения типичной является ситуация, когда необходимо получить некоторые оценки качества рубрикации, которые можно будет использовать для сравнения различных методов и оптимизации параметров метода.

На две части разбивают коллекцию отрубрицированных документов для оценки качества: тестовое, то есть проверочное множество и обучающее, то есть тренировочное множество. Обучение алгоритм осуществляется на тренировочном множестве. Обученный алгоритм применяют к тестовому множеству и рассчитывают на его основе метрики качества рубрицирования [110].

Качество рубрицирования зависит от того, каким образом было осуществлено разбиение множества отрубрицированных документов на тестовое и обучающее множество. В данном случае стоит учитывать пару моментов:

- чем больше обучающее множество, тем лучше можно обучить алгоритм, однако, на малом тестовом множестве оценки качества могут быть слишком грубыми;
- специально подобранное разбиение отрубрицированных документов может оказать влияние на полученный итог и привести, либо к повышению, либо к понижению оценок качества [111].

Качество выстроенного классификатора оценивается при помощи его ошибки на тестовом подмножестве обучающего множества документов. Под ошибкой подразумевается доля неправильно принятых решений классификатором. Получившиеся решения классификатора сравнивают с решениями экспертов, которые формируют обучающее множество.

Статистические величины, используемые для оценки эффективности построенного классификатора, вычисляются следующим образом [112 с. 4]:

Вычисление полноты $r(u)$ (*recall*) классификации информации по классам осуществляется как: «отношение количества документов, правильно приписанных к классу к общему количеству документов, относящихся к данному классу» [112 с.4]:

$$r(u) = \frac{|u \cap v|}{|v|}, \tag{2.7}$$

где

v – множество документов, принадлежащих классу,

u – множество документов, приписанных классу алгоритмом.

Точность $p(u)$ (*precision*) классификации информации по классам вычисляется как: «отношение количества документов, правильно приписанных к классу к общему количеству документов, приписанных к данному классу» [112 с. 4]:

$$p(u) = \frac{|u \cap v|}{|u|}, \tag{2.8}$$

где

v – множество документов, принадлежащих классу,

u – множество документов, приписанных классу алгоритмом.

Объединив оценки полноты и точности в одну, получим метрику качества, называемую *F*–мера (*F–measure*) [112 с. 4]:

$$F(u) = \frac{2 * p(u) * r(u)}{p(u) + r(u)}. \tag{2.9}$$

Если $p(u) = 0$ или $r(u) = 0$, то $F(u) = 0$.

Макроусреднение характеристик по всем классам вводится для получения сводных оценок качества классификации в целом.

$$Macro - p = \frac{1}{|C|} \sum_{i=1}^{|C|} p(u_i), \tag{2.10}$$

$$Macro - r = \frac{1}{|C|} \sum_{i=1}^{|C|} p(u_i), \tag{2.11}$$

$$Macro - F = \frac{1}{|C|} \sum_{i=1}^{|C|} p(u_i). \tag{2.12}$$

При условии произведения классификации документов по нескольким классам, для получения сводных оценок метрик качества используются разные методы усреднения характеристик по всем классам. Отдельная задача заключается в выборе метода усреднения. Пара наиболее часто используемых методов усреднения приведены далее: *microaverage* и *macroaverage* [5 с. 312].

Допустим к каждому классу $C_1, \dots, C_n$ автоматически присвоены документы $u_1, \dots, u_n$. Следовательно, сводные оценки точности и полноты можно определить, как [4 с. 257]:

$$p_{macroavg} = \frac{1}{n}\sum_{i=1}^{n}\frac{|u_i \cap C_i|}{|u_i|}, r_{macroavg} = \frac{1}{n}\sum_{i=1}^{n}\frac{|u_i \cap C_i|}{|C_i|} \text{(макроусреднение)}, \quad (2.13)$$

$$p_{macroavg} = \sum_{i=1}^{n}\frac{|u_i \cap C_i|}{|u_i|}, r_{macroavg} = \sum_{i=1}^{n}\frac{|u_i \cap C_i|}{|C_i|} \text{ (микроусреднение)}, \quad (2.14)$$

где

C_i – количество рубрик,

u_i – документы, приписанные к этим рубрикам.

Микроусредненные и макроусредненные можно определить аналогично оценки *F*. Макроусреднение используется более часто, так как отражает поведение метода в среднем по рубрикам [4 с. 257].

Оценка эффективности классификаторов документов скорее носит экспериментальный характер, чем аналитический. Причиной тому является неформализованность и субъективность задачи текстовой классификации. Поэтому при экспериментальной оценке классификаторов обычно определяют не сложность алгоритма классификатора, а его эффективность, то есть способность правильно распределять документы по категориям.

2.6 Практическое сравнение методов машинного обучения

Значительная часть исследований эффективности, методов автоматической классификации информации, осуществляется на популярной коллекции финансовых сообщений информационного агентства Рейтер – Reuters–21578 [113, 114], создание которой было направлено на исследование эффективности методов автоматической систематизации текстов. Данная коллекция обладает следующими особенностями:

- информация сообщений обладает небольшим по величине размером и относится к узкой предметной области биржевых и финансовых новостей;
- классификатор достаточно прост и состоит из 135 рубрик, без иерархий, как правило, для тестирования используется лишь 10 наиболее часто используемых рубрик [115];
- присвоение тематик рубрикам осуществлялось под контролем качества работы экспертов.

В частности, 40% из имеющихся 21578 документов не рекомендуются к использованию из-за того, что определение классов у них признано некачественным. Оставшиеся 12902 документа отмечаются как «качественно отрубрицированные» [116, 117].

Результаты применения машинного обучения для 10 наиболее частотных рубрик коллекции Reuters–21578 достаточно высоки – примерно около 84% *F*–меры. Сравнение существующих исследований эффективности методов машинного обучения на коллекции Reuters–21578 [118; 119 с. 30] показали, что наибольшую эффективность продемонстрировал метод опорных векторов *SVM* по сравнению с методами деревьев решений, Байеса, нейронных сетей, ближайших соседей, C 4.5, байесовских сетей и *Rocchio* [120].

Анализ опубликованных работ позволяет сделать вывод о том, что:

- большая часть методов, прошедших тестирование на коллекции Reuters–21578, состоит из коротких сообщений с достаточно простым классификатором;
- для проведения тестирования отбираются только классы с большим количеством примеров.

Применение представленных в литературе методов машинного обучения неэффективно, что является значимым для практических задач. Для реальных задач все также достаточно часто применяется ручной труд экспертов, а также системы классификации, основанные на вручную задаваемых правилах [54 с. 34].

На основании проведенного выше анализа можно сделать выводы о том, что наиболее быстрым является метод *SVM*.

Использование векторно-пространственной модели представления текста не лишено недостатков. Факторами, делающими невозможным или усложняющими применение методов машинного обучения для автоматической классификации информации, являются следующие [121]:

- пренебрежение простейшей дополнительной обработкой, такой как морфологический анализ, значительно снижает качество работы классификатора, так как разные формы одного и того же слова считаются разными терминами, вместе с тем морфологический анализ является весьма нетривиальной задачей, требующей для своего решения привлечение лингвистов;
- размерность векторов признаков непосредственно зависит от общего количества терминов в обучающей выборке, текстовых документов, что в реальных задачах приводит к необходимости разрабатывать альтернативные структуры данных, отличные от векторов;
- в словарь терминов могут не входить все документы, подлежащие классификации, так что анализируемые документы могут содержать значимые термины, не вошедшие в обучающую выборку, что отрицательно сказывается на адекватности работы модели.

Создание достаточно большой, последовательно отклассифицированной текстовой коллекцией является серьезной организационной проблемой.

Выше были представлены проблемы, возникающие при решении задач автоматической классификации текстовых документов, а также проанализированы недостатки и преимущества основных методов классификации текста [122].

Некоторые алгоритмы обучения делают определенные предположения о структуре данных и желаемых результатах. Если определить алгоритм,

соответствующий потребностям, то его использование уменьшит время обучения и получения более точных результатов и прогнозов (Таблица 2.5) [123, 124].

Таблица 2.5 – Сравнительный анализ алгоритмов машинного обучения

Название алгоритма	Вычислительная сложность		Скорость обучения	Точность тестирования	Настройка параметров	Эффективность
	Обучение	Тестирование				
1	2	3	4	5	6	7
Naive Bayes	$O(\|\Omega\|)$	$O(\|\Omega\|)$	Быстрая	Хорошая	В зависимости от полученных данных создаются и задаются параметры.	Оптимален для широкого класса задач. Берет во внимание лишь индивидуальное влияние входных переменных.
SVM	$O(\|C\|\|\Omega\|^2)$	$O(\|C\|)$	Низкая	Хорошая	Минимум	Особенно полезен при больших наборах данных.
k–NN	$O(\|\Omega\|)$	$O(\|C\|)$	Низкая	Хорошая	Минимум	При наличии шума в наборе данных значительно искажается результат.
Decision trees	$O(\|\Omega\| log\|\Omega\|)$	–	Низкая	Высокая	Несколько. Параметры создаются и задаются программистом в зависимости от полученных данных	Даёт высокую результативность в вопросах, на которые нужен однозначный логический ответ.

Проведенный анализ позволил сделать следующие выводы:

- для улучшения характеристик классификатора и систематизации текста необходимо последовательное объединение нескольких алгоритмов классификации;
- при формировании последовательности необходимо оптимизировать критерии качественного обучения каждого метода классификации;
- для увеличения точности классификации при объединении методов и взаимодействии между собой необходимо их оптимальное сочетание.

2.7 Выводы к главе 2

В данной главе проведен анализ основных подходов обработки текстовой информации и применения общепринятых методов оценки результатов классификации, проанализированы базовые технологии машинного обучения и лингвистические процессы естественного языка.

1. Обоснован выбор методов для построения модели модернизированной системы систематизации и управления текстовой информацией.
2. Анализ современных публикаций позволяет утверждать, что существует значительный разрыв между методами систематизации и управления информацией основанными на машинном обучении и методами основанными на знаниях.
3. Для дальнейших исследований проведён анализ и выбран один из наиболее эффективных методов систематизации и управления информацией, а именно *SVM*. Использование метода *SVM* станет отправной точки для сравнения качества работы с другими модернизированными моделями, обрабатывающими специализированную текстовую информацию.

ГЛАВА 3
ФОРМАЛИЗАЦИЯ ПРЕДСТАВЛЕНИЯ И ТЕОРЕТИЧЕСКИЕ ОСНОВЫ РАЗРАБОТКИ УСОВЕРШЕНСТВОВАННОЙ ОБОБЩЕННОЙ МОДЕЛИ И АРХИТЕКТУРЫ АНАЛИЗА И УПРАВЛЕНИЯ ИНФОРМАЦИЕЙ

В данной главе выполнена разработка обобщенной модернизированной модели систематизации и управления специализированной информацией.

Ее использование обеспечит повышение эффективности систематизации, управления и своевременный доступ к актуальной информации больших объёмов. Создание модели позволяет отобразить работу реальной системы систематизации и управления специализированной информацией и исследовать ее функциональные характеристики. Благодаря применению обобщенной модели, появляется возможность оценить систему в состоянии равновесия и степень её чувствительности к различным внешним воздействиям, а также исследовать устойчивость поведения полученной структуры, способствующей обработке текстовой информации.

Далее, создание улучшенной основы представления знаний и управления ими заключаются в конструировании множества имитационных моделей, описывающих влияние того или иного воздействия на поведение системы. Отметим, что в данных исследованиях не преследуется цель создания некой «супермодели». Речь идет о разработке частных моделей, которые решают только им присущие особенности систематизации документов. Для облегчения ориентации пользователя в пространстве большого массива информации предлагается использование оптимального способа систематизации и управления знаниями, представляемого в виде стандарта *IDEF*. Формализация данных единым универсальным системным способом представления знаний позволяет усовершенствовать соответствующие алгоритмы и подобрать инструментальные средства для обработки знаний с помощью единого формального аппарата [39 с. 14].

3.1 Математическая модель классификатора для формальной постановки задачи систематизации и управления текстовой информации

Математическая модель классификатора способствует формализации задачи систематизации и управления текстовой информации.

Представим ее следующим образом [125 с. 2]:

Допустим, в наличии имеется множество объектов $T = \{t_i\}$, количество которых не обязательно конечно, а так же множество $C = \{c_i\}\, i = 1..N_c$, состоящее из N_c классов объектов.

Каждый из классов c_i представим при помощи некоторого описания F_i, состоящего из определенной внутренней структуры.

Процесс классификации f объектов $t \in T$ состоит из выполнения трансформации над ними, вследствие которых либо получаем вывод о соответствии t одной из структур F_i, что обозначает присвоение t к классу c_i, или происходит получение вывода о невозможности осуществления классификации t.

Относительно задачи систематизации информации, при помощи элементов множества T используется информация, представленная в виде электронных текстовых документов.

Алгебраическая система служит для представления общей модели текстового классификатора и имеет следующий вид:

$$R = \langle T, C, F, R_c, f \rangle, \tag{3.1}$$

где

T – множество текстовых документов, требующих систематизации,

C – множество тематик – рубрик – классов,

F – множество описаний,

R_c – отношение на $C \times F$,

f – операция классификации, имеющая вид $T \rightarrow C$.

Отношение R_c обладает свойством:

$$\forall c_i \in C \exists F_i \in F : (c_i, F_i) \in R_c, \quad (3.2)$$

другими словами, класс имеет единственное соответствующее ему описание.

Представление f не обладает ограничениями, так что вероятны ситуации, когда:

$$\exists t \in T : f(t) = C_t \subset C \wedge |C_t| > 1, \quad (3.3)$$

то есть некоторая информация случайно может быть отнесена к более чем одному классу одновременно.

Помимо изложенной задачи классификации формируется задача обучения классификатора, которая подразумевает полное или частичное формирование C, F, R_c и f опираясь на некоторые априорные данные [126].

Исходя из представленной формальной постановки задачи систематизации текстовой информации, классификаторы могут быть разделены в зависимости от способа представления описаний классов, а также от организации процедуры классификации.

3.2 Функциональное представление систематизации и управления текстовой информации в *IDEF0*

Используемым стандартом, применяемым для представления сложных систем, является методология *IDEF0*. Применение которой, ориентировано на определение указаний и требований функций, используемых при последующей разработке сложной системы, соответствующей обозначенным требованиям. Итогом использования *IDEF0* в системе становится модель разрабатываемой системы, содержащая в себе иерархически упорядоченный набор текстов и диаграмм, взаимосвязанных при помощи перекрестных ссылок [127].

Диаграммы *IDEF0* строятся при помощи трех наиболее важных типов документов, а именно глоссариев, текстов и графических диаграмм. Документы, используемые для построения диаграмм, включают в себя перекрестные ссылки,

объединяющие между собой работы и отображающие взаимосвязи и взаимодействия между ними [33 с. 12].

Главным компонентом модели *IDEF0* является графическая диаграмма, содержащая стрелки, блоки, соединения стрелок и блоков и ассоциированные с ними отношения. Основные функции моделируемого объекта представляются при помощи блоков. Используемые функции могут быть подвержены разбиению на составные части и представляться при помощи более подробных диаграмм. Для достижения целей определенного проекта необходимо использовать процесс декомпозиции до тех пор, пока объект не достигнет уровня достаточной детализации. Диаграмма верхнего уровня предоставляет абстрактное или общее представление моделируемого объекта. Следом за этой диаграммой идет серия дочерних диаграмм, позволяющих более детально представить изучаемый объект [33 с. 12].

В соответствии с рисунком 3.1 представлены основные исходные данные, используемые для работы системы анализа информации, которые разбиваются на:

- массивы документов для обработки;
- управляющую информацию (методика обучения, параметры моделей классификации, лингвистические словари и т. п.);
- программное обеспечение (библиотеки, системное программное обеспечение).

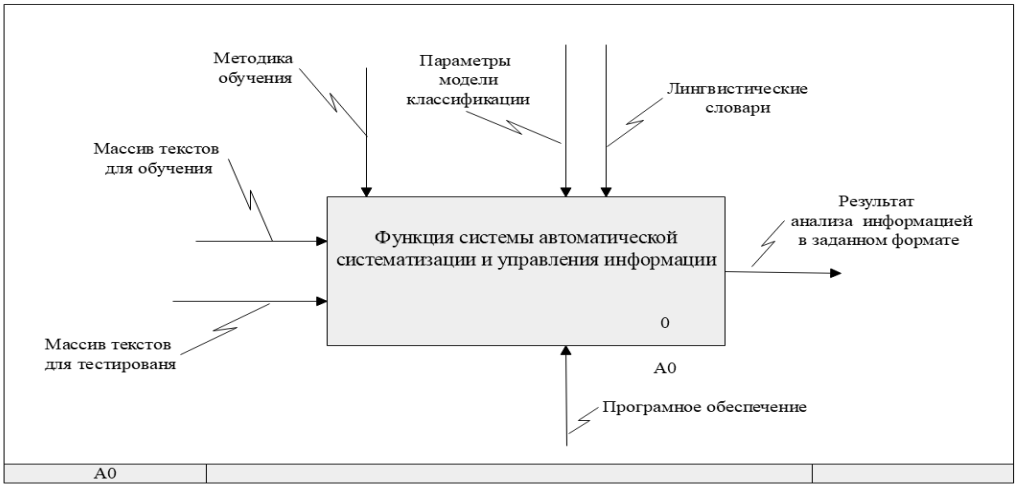

Рисунок 3.1 – *IDEF0* контекстная диаграмма *A*–0

В соответствии с рисунком 3.1, представленная диаграмма называется *A–0* (*A* минус ноль). Связи объекта моделирования с окружающей средой отображены на диаграмме при помощи стрелок. Так как вся модель представлена одним блоком, ее имя становится общим для всей системы систематизации и управления информацией.

Данный факт является справедливым и для стрелок диаграммы, так как они образуют полный комплект интерфейсов модели системы. С помощью диаграммы *A–0* устанавливаются и определяются границы области моделирования. Как правило, важнейшие свойства объекта оказываются на верхних уровнях иерархии; в процессе декомпозиции функции верхнего уровня, а также при разбиении ее на подфункции появляется необходимость в уточнении свойств. Каждая подфункция, в свою очередь, декомпозируется на элементы следующего уровня, и так происходит до тех пор, пока не будет получена релевантная структура, позволяющая достичь поставленных результатов.

Каждая подфункция моделируется как отдельный блок, родительский блок, описание которого развернуто, представлено дочерней диаграммой на более низком уровне. Имеющиеся дочерние диаграммы находятся строго в пределах области контекстной диаграммы верхнего уровня.

В соответствии с рисунком 3.2 представлена дочерняя диаграмма, подвергающая детализации контекстную диаграмму.

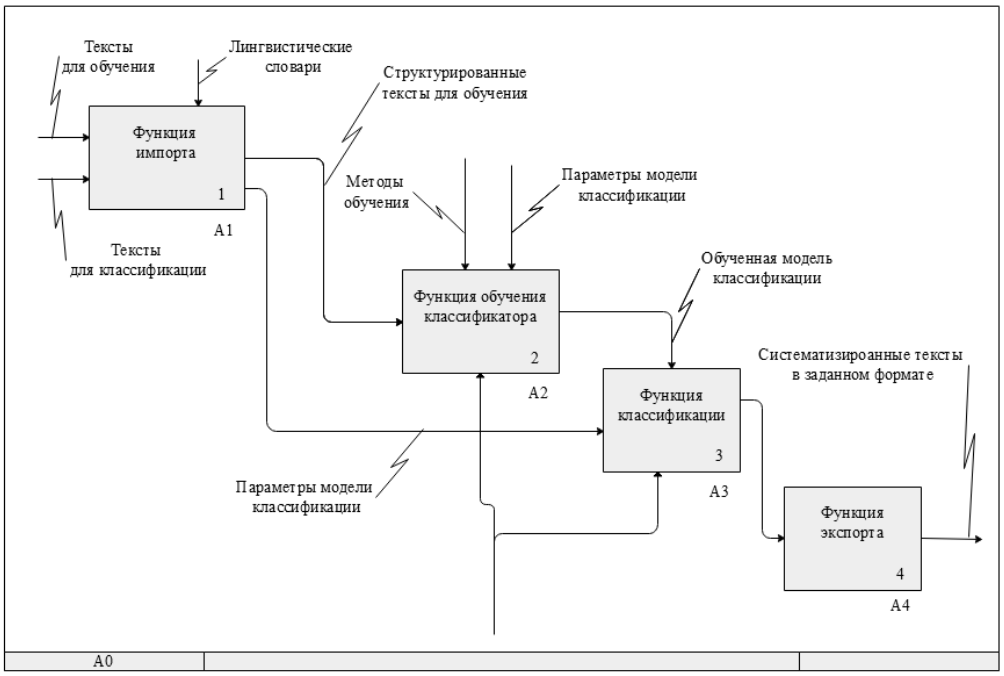

Рисунок 3.2 – *IDEF0* контекстная диаграмма *A–0*. Функции подсистемы автоматической классификации

Каждый блок, изображённый в соответствии с рисунком 3.2, представлен функцией. Описанная диаграмма верхнего уровня, разделена на основные подфункции с помощью формирования дочерней диаграммы. Каждая из этих подфункций, в свою очередь, может быть разделена на составные части средствами построения дочерней диаграммы, более низкого уровня, на котором функции аналогично разложены на составные части. Дополнительная детализация родительского блока обеспечивается дочерней диаграммой, содержащей дочерние стрелки и блоки.

Дочерняя диаграмма, создаваемая при декомпозиции, охватывает ту же область, что и родительский блок, но описывает ее более подробно. Таким образом, дочерняя диаграмма вложена в свой родительский блок [33 с. 12].

Так блок, изображенный в соответствии с рисунком 3.2 диаграмма *A–1* «Функция импорта» становится вложенным блоком для диаграммы следующего, более низкого уровня в соответствии с рисунком 3.3, в которой функции также разложены на составные части, представляющие собой технологию подготовки документов для обработки информации.

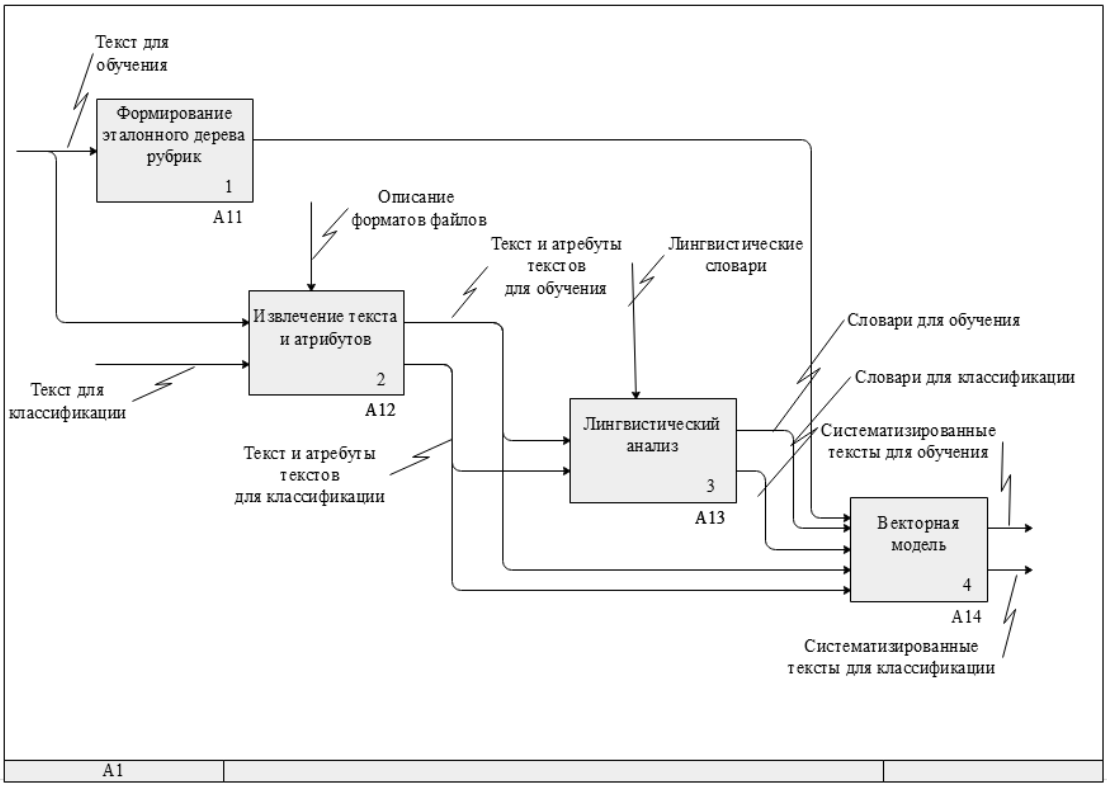

Рисунок 3.3 – *IDEF0* контекстная диаграмма *A–1*. Функция импорта

Обработка информации, при помощи предполагаемых композиций алгоритмов, способствующих уменьшению размера документов, а также методы преобразования текста в вектор более подробно будут описаны в разделе 3.3.

В соответствии с рисунком 3.4 представлены основные операции работы обучающего классификатора, представление которого осуществлено в разделе 3.4.

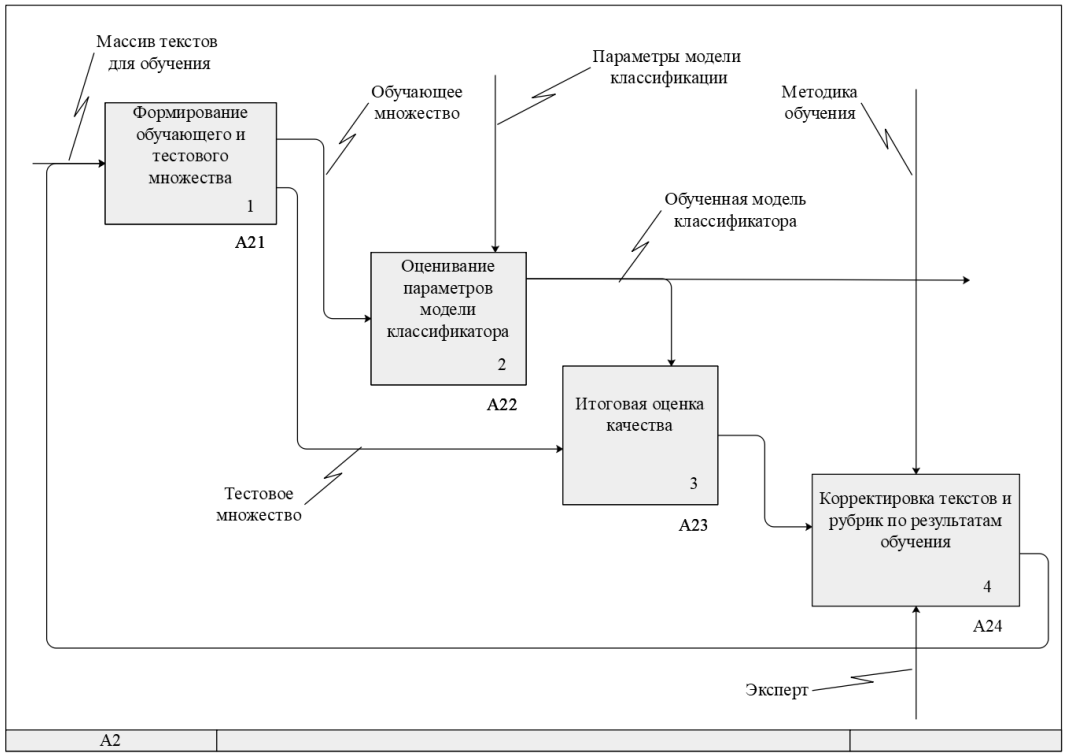

Рисунок 3.4 – *IDEF0* контекстная диаграмма *A*–2. Функция обучения классификатора

Необходимость выполнения такой коррекции объясняется тем, что в процессе ручной классификации часто возможны ошибки как связанные с ошибочным отнесением фрагментов текстов к нерелевантным рубрикам, так и к пропуску некоторых релевантных рубрик. Формируемый отчет позволяет выделить такого рода тексты путем анализа списков пропущенных и добавленных документов. Для обучения классификатора в системе поддерживается методика, описанная в разделе 3.4.

В соответствии с рисунком 3.5 представлен в общем виде процесс, выполняемый при использовании классификатора.

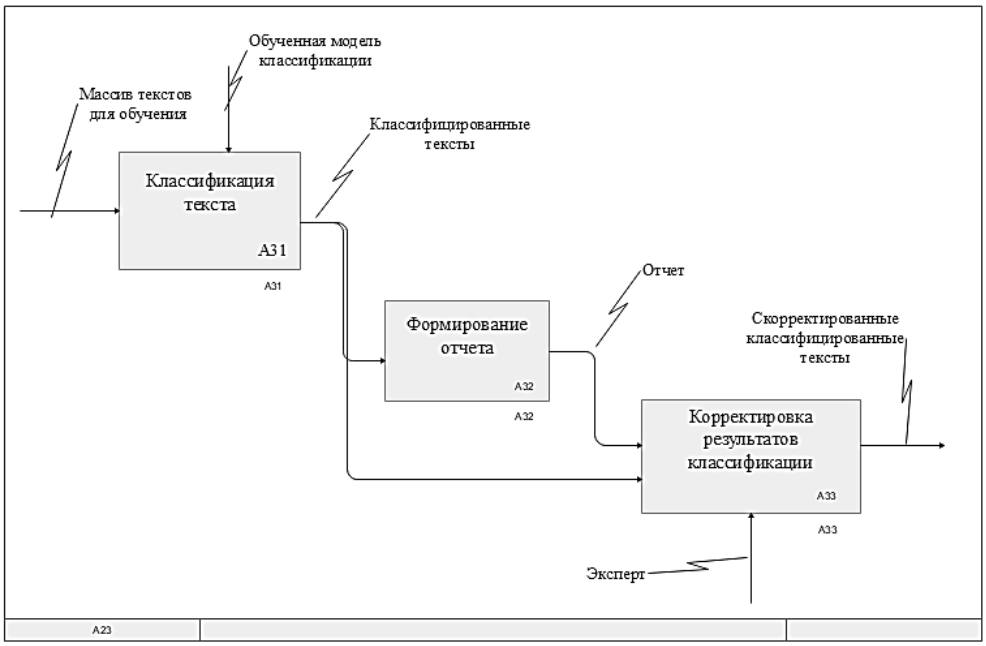

Рисунок 3.5 – *IDEF0* контекстная диаграмма *A–3*. Функция классификации текста

В качестве метода классификации использована модель алгоритма, при работе которого будет достигаться оптимальная эффективность его работы. Разработанные и модернизированные композиции алгоритмов классификации более подробно будут представлены в разделе 3.5 [3 с. 275].

Определение эффективности итоговой модели используемой для систематизации текста, то есть способной правильно распределять документы по тематикам, будет определяться при помощи метрик качества, включающих в себя полноту, точность и *F*–меру. Формулы расчета метрик качества работы полученной модели, более подробно представлены в разделе 3.6 [5 с. 71].

В теории, функциональная модель обработки информации должна быть точной, понятной и доступной для использования. Но на практике при построении такой модели разработчики сталкиваются с рядом трудностей, а потому в процессе изучения и поиска решения поставленной задачи создается функциональная модель работы. Графическое представление создаваемой модели, позволяет понимать, как работает та или иная система и можете также наглядно пояснить, где в этой системе «тонкие места», и как решения могут помочь избавиться от них.

В процессе исследования возникает необходимость не просто изучить и решить определенную проблему, возникающую при систематизации текстовой информации, но выявить ее местонахождение в общей модели. Важно понимать, каким образом возникающие сложности взаимодействуют с другими блоками модели. Иначе невозможно выявить все существующие проблемы и выбрать оптимальный метод решения поставленной задачи. Для этого требуется изучить работу различных функциональных схем, управляющих информацией и составить функциональную модернизированную модель системы [128].

3.3 Предобработка и векторизация специализированной информации

Представим более подробно некоторые из методов, способствующие уменьшению размера текстового документа. Каждый из них будет использоваться для нахождения и построения оптимальной композиции, предварительной обработки текста:

1. Метод «Стоп–слова» («Шумовые слова») – это слова, являющиеся вспомогательными, которые несут малую смысловую нагрузку о содержании документа. Для выявления таких слов будет использоваться заранее составленный список по каждому корпусу текстов вручную. В процессе предварительной обработки «шумовые–слова» удаляются из текста. К стоп–словам относят предлоги, союзы, местоимения и так далее приведенные в приложениях А и Б [129].
2. Метод «Стемминг» – морфологический поиск, который заключается в преобразовании каждого слова и приведении его к общей форме. В этом подходе главным образом используется отбрасывание окончаний в словах. Эта технология алгоритма морфологического разбора, учитывающая языковые особенности, вследствие чего данный алгоритм является языково-зависимым [130].

Но классификаторы не работают с буквенными текстами, для их работы используется вектор. Другими словами, словам в текстовом документе

необходимо присвоить некоторые числовые значения. Наиболее распространенным методом построения вектора является метод *TF–IDF*.

Данный метод определяет вес некоторого слова, который «пропорционален количеству употребления его в документе и обратно пропорционален частоте употребления слова в других документах коллекции». Вес слова вычисляется по формуле [131]:

$$TF - IDF(w, d, D) = TF(w, d) \times IDF(w, D), \quad (3.4)$$

где

TF (*term frequency* – частота терма) – «отношение числа вхождения некоторого терма w к общему количеству термов документа. Таким образом, оценивается важность терма w в пределах отдельного документа d. Частота слова оценивает важность слова w_i в пределах отдельного документа» [132]:

$$TF(w, d) = \frac{n_i}{\sum_k n_k}, \quad (3.5)$$

где

n_i – число вхождений слова i в документ,

$\sum_k n_k$ – общее число слов в данном документе.

Инверсия частоты *IDF* (*inverse document frequency*) – «обратная частота документа, с которой некоторое слово встречается в документах коллекции. Учёт *IDF* уменьшает вес широкоупотребительных слов. Для каждого уникального слова в пределах конкретной коллекции документов существует только одно значение *IDF*» [133]:

$$IDF(w, D) = \frac{|D|}{|(d_i \supset w_i)|}, \quad (3.6)$$

где

$|D|$ – количество документов в корпусе,

$|(d_i \supset w_i)|$ – количество документов, в которых встречается слово w_i.

В таблице 3.1 приведены предлагаемые дополняющие друг друга комбинации методов, используемые для уменьшения размера текстовых документов, которые в дальнейшем будут преобразованы в вектор. Выбор был остановлен именно на этих методах, так как удаление неинформативных слов и окончаний делают его частично независимым от тематики текстовых документов.

Таблица 3.1 – Композиции методов уменьшения размера текста

Сокращение	Композиция методов	Краткое описание
1	2	3
Т–I	$TF - IDF(w, d, D)$	Использование метода $\mathrm{TF} - \mathrm{IDF}$ для преобразования текста в вектор без какой–либо дополнительной обработки. При расчете используется классическая формула: $\mathrm{TF} - \mathrm{IDF}(\mathrm{w}, \mathrm{d}, \mathrm{D}) = \frac{\mathrm{n_i}}{\sum_k \mathrm{n_k}} \times \frac{\lvert \mathrm{D} \rvert}{\lvert (\mathrm{d_i} \supset \mathrm{w_i}) \rvert}$.
Т–I +Ш	$TF - IDF(w, d, D)$+ «Шумовые слова»	Удаление слов, не несущих смысловой нагрузки.
Т–I +С	$TF - IDF(w, d, D)$+ «Стемминг»	Приведение слов в тексте к единой основе.
Т–I +С+Ш	$TF - IDF(w, d, D)$+ «Стемминг» + «Шумовые слова»	Приведение слов в тексте к единой основе. Удаление шумовых слов, не несущих в себе смысловую нагрузку текста.

В таблице 3.1 представлены модернизированные композиции, состоящие из нескольких методик. Удаление шумовых слов позволит уменьшить размер документа, что повысит скорость работы алгоритма. А удаление окончаний позволит электронной вычислительной машине корректно распределить вес слов, что в дальнейшем повысит качество работы классификатора.

Следовательно, применение композиций, состоящих из дополнительных методов обработки текстовой информации, влияет на качество распределения

веса слов, осуществляемое путем использования основной формулы *TF–IDF*. Разработанные и предлагаемые мною композиции представлены в виде формул, условные обозначения которых взяты из таблицы 3.1:

1. $(w, d, D) = TF(w, d) \times IDF(w, D)$,
2. $TF - IDF(w, d, D) + \text{Ш} = \left(\frac{n_i}{\sum_k n_k} \times \frac{|D|}{|(d_i \supset w_i)|}\right) + \text{Ш}$,
3. $TF - IDF(w, d, D) + \text{С} = \left(\frac{n_i}{\sum_k n_k} \times \frac{|D|}{|(d_i \supset w_i)|}\right) + \text{С}$,
4. $TF - IDF(w, d, D) + \text{С} + \text{Ш} = \left(\frac{n_i}{\sum_k n_k} \times \frac{|D|}{|(d_i \supset w_i)|}\right) + \text{С} + \text{Ш}$.

Также вес некоторых слов можно увеличить при помощи вариаций коэффициентов *L*, *Q* и *Z*.

Коэффициент *L* представляет собой:

$$L = \frac{n_i}{|(d_i \supset w_i)|}, \tag{3.7}$$

где

n_i– число вхождений слова *i* в документ,

$|(d_i \supset w_i)|$– количество документов, в которых встречается слово w_i.

Под коэффициентом *Q* будем понимать словарь терминов, используемый только для конкретной рубрики, и при нахождении данного терма в текстовом документе его вес будет дополнительно увеличен.

По такому же принципу будет действовать и коэффициент *Z*, но разница состоит в том, что в данный словарь войдут слова, входящие в название класса.

Таким образом, в формуле определения веса будет варьироваться в зависимости от внесённых в нее изменений. Предложенные изменения будут вноситься в два этапа, и выглядеть представленным далее образом.

Первый этап рассматривает влияние коэффициента *L*, на качество классификации, при использовании вариаций распределения веса слов в документе:

1. $$TF-IDF(w,d,D)+\text{С}+\text{Ш}=\left(\frac{n_i}{\sum_k n_k}\times\frac{|D|}{|(d_i\supset w_i)|}\right)+\text{С}+\text{Ш},$$

2. $$TF-IDF(w,d,D)=\frac{n_i+L}{\sum_k n_k}\times\frac{|D|}{|(d_i\supset w_i)|}+\text{С}+\text{Ш}=\frac{n_i+\frac{n_i}{|(d_i\supset w_i)|}}{\sum_k n_k}\times$$
$$\times\frac{|D|}{|(d_i\supset w_i)|}+\text{С}+\text{Ш},$$

3. $$TF-IDF(w,d,D)=\frac{n_i*L}{\sum_k n_k}\times\frac{|D|}{|(d_i\supset w_i)|}+\text{С}+\text{Ш}=\frac{n_i*\frac{n_i}{|(d_i\supset w_i)|}}{\sum_k n_k}\times$$
$$\times\frac{|D|}{|(d_i\supset w_i)|}+\text{С}+\text{Ш},$$

4. $$TF-IDF(w,d,D)=\frac{n_i}{\sum_k n_k}\times\frac{|D|}{|(d_i\supset w_i)|}+L+\text{С}+\text{Ш}=\frac{n_i}{\sum_k n_k}\times$$
$$\times\frac{|D|}{|(d_i\supset w_i)|}+\frac{n_i}{|(d_i\supset w_i)|}+\text{С}+\text{Ш},$$

5. $$TF-IDF(w,d,D)=\frac{n_i}{\sum_k n_k}\times\frac{|D|}{|(d_i\supset w_i)|}*L+\text{С}+\text{Ш}=\frac{n_i}{\sum_k n_k}\times$$
$$\times\frac{|D|}{|(d_i\supset w_i)|}*\frac{n_i}{|(d_i\supset w_i)|}+\text{С}+\text{Ш}.$$

Таким образом, были получены композиции методов, определяющие вес термов в документе, представленные в таблице 3.2.

Таблица 3.2 – Композиции методов векторного представления текста на первом этапе влияния коэффициента L, на качество классификации

Сокращение	Композиция методов	Краткое описание
1	2	3
Т–I+С+Ш	$TF-IDF(w,d,D)$+ «Стемминг» + «Шумовые слова»	При приведении слов в тексте к единой основе, удаление шумовых слов, не несущих в себе смысловую нагрузку текста используем следующую формулу: $TF-IDF(w,d,D)+\text{С}+\text{Ш}=$ $=\left(\frac{n_i}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}\right)+\text{С}+\text{Ш}.$
((Т–I)+С+Ш)+L– 1	«Доп. коэффициент L» + $TF-IDF(w,d,D)$+ «Стемминг» +	Использование коэффициента L для изменения веса слов. Приведение текста к векторному виду по формуле:

Продолжение таблицы 3.2.

Сокращение	Композиция методов	Краткое описание
1	2	3
	«Шумовые слова»	$TF-IDF(w,d,D)=\frac{n_i+L}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}-\text{С}-$ $-\text{Ш}=\frac{n_i+\frac{n_i}{\lvert(d_i\supset w_i)\rvert}}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+\text{С}+\text{Ш}.$
((Т–І)+С+Ш)+L– 2	«Доп. коэффициент L» + $TF-IDF(w,d,D)$ + «Стемминг» + «Шумовые слова»	Использование коэффициента L для изменения веса слов. Приведение текста к векторному виду по формуле: $TF-IDF(w,d,D)=\frac{n_i*L}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+\text{С}+$ $+\text{Ш}=\frac{n_i*\frac{n_i}{\lvert(d_i\supset w_i)\rvert}}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+\text{С}+\text{Ш}.$
((Т–І)+С+Ш)+L–3	«Доп. коэффициент L»+ $TF-IDF(w,d,D)$ + «Стемминг» + «Шумовые слова»	Использование коэффициента L для изменения веса слов. Приведение текста к векторному виду по формуле: $TF-IDF(w,d,D)=\frac{n_i}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+L+$ $+\text{С}+\text{Ш}=\frac{n_i}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+\frac{n_i}{\lvert(d_i\supset w_i)\rvert}+\text{С}+$ $+\text{Ш}.$
((Т–І)+С+Ш)+L–4	«Доп. коэффициент L»+ $TF-IDF(w,d,D)$ + «Стемминг» + «Шумовые слова»	Использование коэффициента L для изменения веса слов. Приведение текста к векторному виду по формуле: $TF-IDF(w,d,D)=\frac{n_i}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}\times L+$ $+\text{С}+\text{Ш}=\frac{n_i}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}\times\frac{n_i}{\lvert(d_i\supset w_i)\rvert}+\text{С}+$ $+\text{Ш}.$

В результате предложены композиции методов, дополнительно влияющие на определение веса термов и представляющие собой заключительный этап построения вектора.

Второй этап рассматривает влияние предложенных коэффициентов Q и Z, на качество классификации, при использовании вариаций распределения веса слов в документе:

1. $TF-IDF(w,d,D)=\frac{n_i}{\sum_k n_k}\times\frac{|D|}{|(d_i\supset w_i)|}+L+\text{С}+\text{Ш}=\frac{n_i*L}{\sum_k n_k}\times$

$\times\frac{|D|}{|(d_i\supset w_i)|}+\frac{n_i}{|(d_i\supset w_i)|}+\text{С}+\text{Ш},$

2. $TF-IDF(w,d,D)=\frac{n_i+Q+Z}{\sum_k n_k}\times\frac{|D|}{|(d_i\supset w_i)|}+L+\text{С}+\text{Ш}=\frac{n_i+Q+Z}{\sum_k n_k}\times$

$\times\frac{|D|}{|(d_i\supset w_i)|}+\frac{n_i}{|(d_i\supset w_i)|}+\text{С}+\text{Ш},$

3. $TF-IDF(w,d,D)=\frac{n_i}{\sum_k n_k}\times\frac{|D|}{|(d_i\supset w_i)|}+L+\text{С}+\text{Ш}+Q+Z=$

$=\frac{n_i}{\sum_k n_k}\times\frac{|D|}{|(d_i\supset w_i)|}+\frac{n_i}{|(d_i\supset w_i)|}+\text{С}+\text{Ш}+Q+Z,$

4. $TF-IDF(w,d,D)=\frac{n_i}{\sum_k n_k}\times\frac{|D|}{|(d_i\supset w_i)|}+L+\text{С}+\text{Ш}=\frac{n_i+Q+Z}{\sum_k n_k}\times$

$\times\frac{|D|}{|(d_i\supset w_i)|}+\frac{n_i+Z+Q}{|(d_i\supset w_i)|}+\text{С}+\text{Ш},$

5. $TF-IDF(w,d,D)=\frac{n_i+Z+Q}{\sum_k n_k}\times\frac{|D|}{|(d_i\supset w_i)|}+L+\text{С}+\text{Ш}=\frac{n_i+Q+Z}{\sum_k n_k}\times$

$\times\frac{|D|}{|(d_i\supset w_i)|}+\frac{n_i+Z+Q}{|(d_i\supset w_i)|}+\text{С}+\text{Ш},$

6. $TF-IDF(w,d,D)=\frac{n_i+Q}{\sum_k n_k}\times\frac{|D|}{|(d_i\supset w_i)|}+L+\text{С}+\text{Ш}=\frac{n_i+Q}{\sum_k n_k}\times$

$\times\frac{|D|}{|(d_i\supset w_i)|}+\frac{n_i}{|(d_i\supset w_i)|}+\text{С}+\text{Ш},$

7. $TF-IDF(w,d,D)=\frac{n_i}{\sum_k n_k}\times\frac{|D|}{|(d_i\supset w_i)|}+L+\text{С}+\text{Ш}+Q+Z=$

$=\frac{n_i}{\sum_k n_k}\times\frac{|D|}{|(d_i\supset w_i)|}+\frac{n_i}{|(d_i\supset w_i)|}+\text{С}+\text{Ш}+Q.$

Таким образом, были получены композиции методов, определяющие вес термов в документе, представленные в таблице 3.3.

Таблица 3.3 – Композиции методов векторного представления текста на втором этапе влияния коэффициентов Q и Z, на качество классификации

Сокращение	Композиция методов	Краткое описание
1	2	3
((Т–I)+С+Ш)+ +L–3	«Доп. коэффициент L» + $TF-IDF(w,d,D)$+ «Стемминг» + «Шумовые слова»	Использование коэффициента L для изменения веса слов. Приведение текста к векторному виду по формуле: $TF-IDF(w,d,D)=\frac{n_i}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+L+$ $+\text{С}+\text{Ш}=\frac{n_i\times L}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+\frac{n_i}{\lvert(d_i\supset w_i)\rvert}+\text{С}+\text{Ш}.$
(((Т–I)+С+Ш)+ +L–3)+Q+Z–1	«Доп. коэффициент Q» + «Доп. коэффициент Z» + «Доп. коэффициент L» + $TF-IDF(w,d,D)$ + «Стемминг» + «Шумовые слова»	Использование коэффициентов L, Q и Z для изменения веса слов. Приведение текста к векторному виду по формуле: $TF-IDF(w,d,D)=\frac{n_i+Q+Z}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+L+$ $+\text{С}+\text{Ш}=\frac{n_i+Q+Z}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+\frac{n_i}{\lvert(d_i\supset w_i)\rvert}+\text{С}+$ $+\text{Ш}.$
(((Т–I)+С+Ш)+ +L–3)+Q+Z–2	«Доп. коэффициент Z» + «Доп. коэффициент Q» + «Доп. коэффициент L» + $TF-IDF(w,d,D)$+ «Стемминг» + «Шумовые слова»	Использование коэффициентов L, Q и Z для изменения веса слов. Приведение текста к векторному виду по формуле: $TF-IDF(w,d,D)=\frac{n_i}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+L+$ $+\text{С}+\text{Ш}+Q+Z=\frac{n_i}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+$ $+\frac{n_i}{\lvert(d_i\supset w_i)\rvert}+\text{С}+\text{Ш}+Q+Z.$
(((Т–I)+С+Ш)+ +L–3)+Q+Z–3	«Доп. коэффициент Z» + «Доп. коэффициент Q» + «Доп. коэффициент L» + $TF-IDF(w,d,D)$ + «Стемминг» + «Шумовые слова»	Использование коэффициентов L, Q и Z для изменения веса слов. Приведение текста к векторному виду по формуле: $TF-IDF(w,d,D)=\frac{n_i+Q+Z}{\sum_k n_k}\times$ $\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+\frac{n_i+Z+Q}{\lvert(d_i\supset w_i)\rvert}+\text{С}+\text{Ш}.$

Продолжение таблицы 3.3

Сокращение	Композиция методов	Краткое описание
1	2	3
(((Т–I)+С+Ш)+ +L–3)+Q+Z–4	«Доп. коэффициент *Z*» + «Доп. коэффициент *Q*» + «Доп. коэффициент *L*» + $TF-IDF(w,d,D)$ + «Стемминг» + «Шумовые слова»	Использование коэффициентов L, Q и Z для изменения веса слов. Приведение текста к векторному виду по формуле: $TF-IDF(w,d,D)=\frac{n_i+Q+Z}{\sum_k n_k}\times$ $\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+\frac{n_i+Z\times Q}{\lvert(d_i\supset w_i)\rvert}+\text{С}+\text{Ш}.$
(((Т–I)+С+Ш)+ +L–3)+Q–1	«Доп. коэффициент *Q*» + «Доп. коэффициент *L*» + $TF-IDF(w,d,D)$ + «Стемминг» + «Шумовые слова»	Использование коэффициентов L и Q для изменения веса слов. Приведение текста к векторному виду по формуле: $TF-IDF(w,d,D)=\frac{n_i}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+L+$ $+\text{С}+\text{Ш}+Q+Z=\frac{n_i}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+$ $+\frac{n_i}{\lvert(d_i\supset w_i)\rvert}+\text{С}+\text{Ш}+Q+Z.$
(((Т–I)+С+Ш)+ +L–3)+Q–1	«Доп. коэффициент *Q*»+ «Доп. коэффициент *L*» + $TF-IDF(w,d,D)$ + «Стемминг» + «Шумовые слова»	Использование коэффициентов L и Q для изменения веса слов. Приведение текста к векторному виду по формуле: $TF-IDF(w,d,D)=\frac{n_i+Q}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+L+$ $+\text{С}+\text{Ш}=\frac{n_i+Q}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+\frac{n_i}{\lvert(d_i\supset w_i)\rvert}+\text{С}+\text{Ш}.$
(((Т–I)+С+Ш)+ +L–3)+Z–1	«Доп. коэффициент *Z*» + «Доп. коэффициент *L*» + $TF-IDF(w,d,D)$ + «Стемминг» + «Шумовые слова»	Использование коэффициентов L и Z для изменения веса слов. Приведение текста к векторному виду по формуле: $TF-IDF(w,d,D)=\frac{n_i+Z}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+$ $+L+\text{С}+\text{Ш}=\frac{n_i+Z}{\sum_k n_k}\times\frac{\lvert D\rvert}{\lvert(d_i\supset w_i)\rvert}+$ $+\frac{n_i}{\lvert(d_i\supset w_i)\rvert}+\text{С}+\text{Ш}.$

Продолжение таблицы 3.3

(((Т–І)+С+Ш)+ +L–3)+Z–2	«Доп. коэффициент *Z*» + «Доп. коэффициент *L*» + $TF - IDF(w, d, D)$+ «Стемминг» + «Шумовые слова»	Использование коэффициентов L и Z для изменения веса слов. Приведение текста к векторному виду по формуле: $TF - IDF(w, d, D) = \frac{n_i}{\sum_k n_k} \times \frac{\lvert D \rvert}{\lvert (d_i \supset w_i) \rvert} + L +$ $+\text{С} + \text{Ш} + Z = \frac{n_i}{\sum_k n_k} \times \frac{\lvert D \rvert}{\lvert (d_i \supset w_i) \rvert} + \frac{n_i}{\lvert (d_i \supset w_i) \rvert} +$ $+\text{С} + \text{Ш} + Z.$

Были получены композиции методов, при помощи вариаций коэффициентов L, Q и Z, использование которых привело к дополнительному влиянию на определение веса термов, что в свою очередь повлияет на качество работы классификатора.

Таким образом, в подразделе представлены разработанные модернизированные модели анализа информации, состоящие из отобранных актуальных композиций, предположительно обеспечивающих повышение эффективности предварительной обработки текстовой информации [134].

Использование этих композиций при удалении шумовых слов позволит уменьшить размер документа, что в свою очередь повысит скорость работы алгоритма. С другой стороны, удаление окончаний и использование дополнительных коэффициентов (а именно L, Q и Z) позволит электронной вычислительной машине корректно распределить вес слов, что в дальнейшем повысит качество работы классификатора.

Использование предложенных факторов приведет к дополнительному влиянию на качество определения важности слова в тексте, а как следствие и качеству работы всей системы, управления и систематизации информации.

3.4 Обучение классификатора

Следующим блоком предложенной модели обработки текстовой информации является блок обучения классификатора.

Работа данного блока поддерживается при помощи следующего алгоритма [3 с. 275]:

1. Произвести обучение и оценку качества работы классификатора.

2. Проверить общее качество обучения. Если оно окажется приемлемым, то перейти к шагу 6, в противном случае перейти к шагу 3.

3. Выбрать для корректировки рубрики, имеющие низкие значения оценок точности и полноты классификации.

4. Для каждой выбранной рубрики выполнить следующие проверки:

a) если в рубрике много дополнительных документов, и они относятся более чем к двум другим рубрикам, то рубрика недостаточно отделяется от других рубрик и возможны следующие операции по ее корректировке:

i) добавить дополнительные документы в рубрику;

ii) выбрать более подходящие обучающие примеры.

b) Если в рубрике много дополнительных документов из одной или двух других рубрик, то возможны следующие операции по корректировке рубрики:

i) добавить дополнительные документы в рубрику;

ii) объединить пересекающиеся рубрики.

c) Если в рубрике много пропущенных документов, и они относятся более чем к двум другим рубрикам, то возможны следующие операции по корректировке рубрики:

i) выбрать более подходящие обучающие примеры;

ii) удалить рубрику.

d) Если в рубрике много пропущенных документов, но они попадают только в одну или две другие рубрики, то возможны следующие операции по корректировке рубрики:

i) добавить пропущенные документы в другие рубрики;

ii) объединить пересекающиеся рубрики.

е) Если много документов попадает в рубрику «Другие» или оценка адекватности массива отрицательна, то возможна следующая операция по корректировке рубрики:

i) добавить новые обучающие документы, соответствующие стилю и размеру документов, попадающих в рубрику «Другие».

5. Перейти к шагу 1.

6. Завершить процедуру обучения классификатора.

При выполнении классификации массива новых документов создается специальный отчет, содержащий описание распределения текстов по рубрикам, и выполняется выделение в каждом документе наиболее значимых фрагментов. В процессе формирования отчета для каждой рубрики автоматически вызываются процедуры группировки «дубликатов», кластерного анализа и тематического упорядочения документов [3 с. 163].

Использование указанного алгоритма предполагает, что выполнение корректировки рубрик имеет большое значение для повышения качества классификации текстовой информации.

3.5 Модернизация композиций алгоритмов, обрабатывающих специализированную текстовую информацию

За основу при построении модернизированной модели системы систематизации и управления информацией взят метод *support vector machine*, предложенный В. Н. Вапником, являющийся контролируемым алгоритмом обучения и принадлежащий к группе методов детерминистского подхода [95 с. 55].

Алгоритм *support vector machine* – двоичный линейный классификатор, способствующий разделению негативных и позитивных примеров в наборе тестовой информации. Реализация метода осуществляется при помощи поиска оптимальной гиперплоскости, максимально разграничивающей отрицательные от

положительных примеров, обеспечивая границу, разделяющую ближайшие негативы и позитивы.

Основной идеей метода *support vector machine* является преобразование начальных векторов в область более высокой размерности и осуществление поиска разбивающей гиперплоскости с максимальным промежутком в данном пространстве. Пара параллельных гиперплоскостей, выстраиваются с обеих сторон гиперплоскости, разбивающей классы. Разбивающей гиперплоскостью является гиперплоскость, максимизирующая промежуток между двумя параллельными гиперплоскостями. Алгоритм работает с предположением, того что чем больше расстояние или разница между параллельными гиперплоскостями, тем меньше окажется средняя ошибка при классификации [93 с. 15].

Предположим, что точки имеют вид:

$$\{(x_1, c_1), (x_2, c_2), \dots, (x_n, c_n)\}, \tag{3.8}$$

где

c_i принимает значение *1* или *–1*, в зависимости от того, какому классу принадлежит точка x_i,

x_i – это мерный вещественный вектор, обычно нормализованный значениями $[0,1]$ или $[-1,1]$.

Появляется необходимость в нормализации точек, так как точка с значительными отклонениями отличными от средних значений координат точек чрезмерно сильно повлияет на классификатор. Для корректной работы алгоритма *SVM* необходимо чтобы он классифицировал документы таким образом, чтобы для каждого найденного элемента уже был определен класс, которому он принадлежит. Для этого используем разделяющую гиперплоскость, следующего вида [5 с. 223]:

$$w \cdot x - b = 0. \tag{3.9}$$

Для построения оптимальной разделяющей гиперплоскости, используемой в методе *SVM*, основывается на точках, принадлежащих двум разным классам. В соответствии с рисунком 3.6. наиболее приближенные к параллельным гиперплоскостям точки являются опорными векторами.

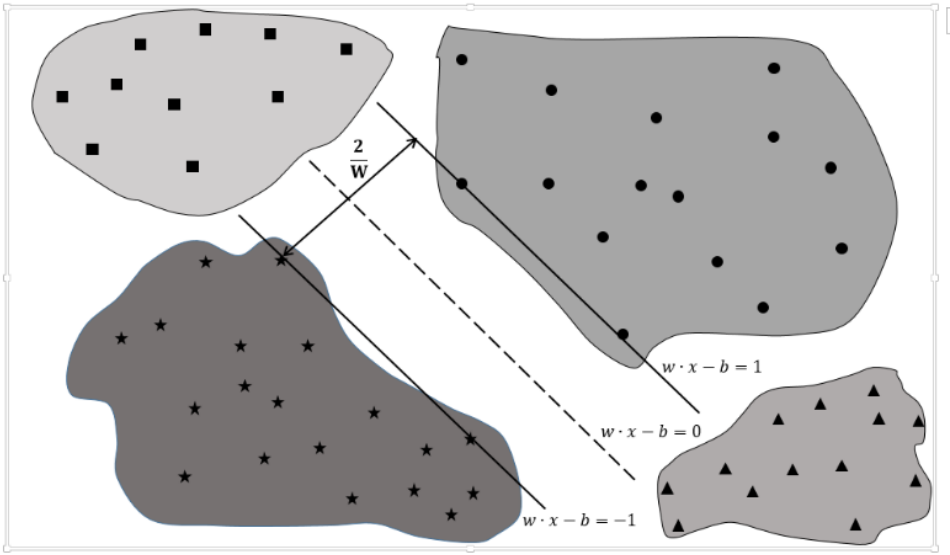

Рисунок 3.6 – Пример оптимального разделения гиперплоскостей для метода опорных векторов

Перпендикуляр к разбивающей гиперплоскости называется вектором. Параметру $\frac{b}{\|w\|}$ присвоено значение по модулю равное расстоянию от гиперплоскости до начала координат. При условии того что параметр b тождественен нулю, гиперплоскость располагается в начале координат, что служит ограничением решения [93 с. 15].

Внимание уделяется оптимальному разделению текстовой информации на классы, которое определяются с помощью построения опорных гиперплоскостей. Полученные параллельные гиперплоскости оказываются ближайшими и оптимальными к опорным векторам некоторых классов. Полученные гиперплоскости можно представить с помощью уравнений [93 с. 15]:

$$\begin{aligned} w \cdot x - b &= 1, \\ w \cdot x - b &= -1. \end{aligned} \tag{3.10}$$

При условии, что обучающая выборка становится линейно разделимой, возникает возможность определения гиперплоскостей с условием отсутствия точек, принадлежащих обучающей выборке, а затем осуществления максимизации расстояния между гиперплоскостями. Ширина полосы между гиперплоскостями определяется по соображениям геометрии, и равна $\frac{2}{\|w\|}$, как следствие возникает задача минимизации $\|w\|$. Для исключения всех точек из полосы, стоит убедиться, в том, что для всех i:

$$\begin{cases} w \cdot x - b \geq 1, c_i = 1 \\ w \cdot x - b \leq -1, c_i = -1 \end{cases}. \quad (3.11)$$

Что может быть представлено в виде:

$$c_i(w \cdot x - b) \geq 1, 1 \leq i \leq n. \quad (3.12)$$

Стоит учитывать то, что представленное изображение в соответствии с рисунком 3.6 является наилучшим. При работе классификатора появляется погрешность, влияющая на качество результата при распределении, то есть отнесении того или иного документа к тому или иному классу. Для повышения качества работы классификатора эту погрешность необходимо минимизировать.

В соответствии с рисунком 3.7 представлено изображение многоклассового классификатора с близкими тематиками. Изображены документы, относящиеся к разным классам, попадающие под ошибочное распределение, для которых линейное разделение принесет значительное ухудшение качества распределения по классам текстовых документов.

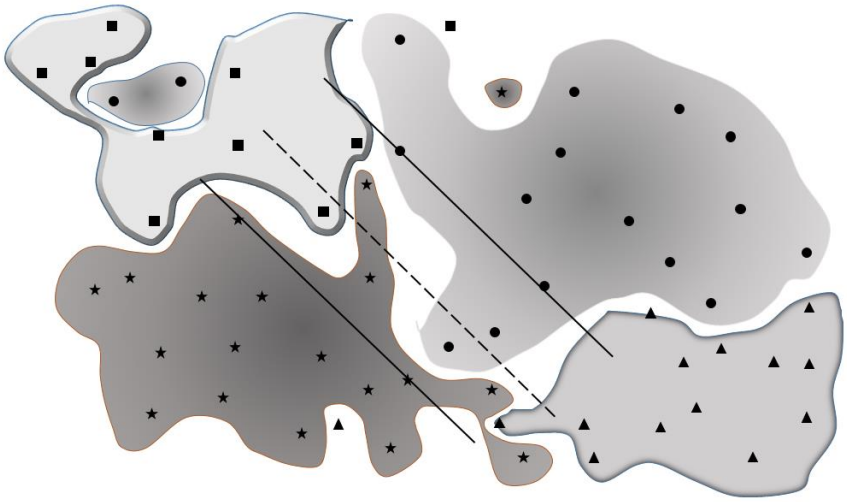

Рисунок 3.7 – Пример многоклассовой классификации.

Для повышения качества классификации необходимы некоторые изменения в порядке определения принадлежности документа к тому или иному классу.

Дополним работу метода опорных векторов другим методом машинного обучения, а именно методом дерева принятия решений.

Использование этого метода позволит модернизировать алгоритм *SVM* в тех случаях, когда основной алгоритм распределит документ одновременно к нескольким тематикам. Использование такой композиции позволит повысить качество систематизации текстовой информации.

Дерево принятия решений состоит из ветвей и листьев. Ветви (ребра графа) хранят в себе значения атрибутов, от которых зависит целевая функция; на листьях же записывается значение целевой функции в соответствии с рисунком 3.8 [101 с. 120].

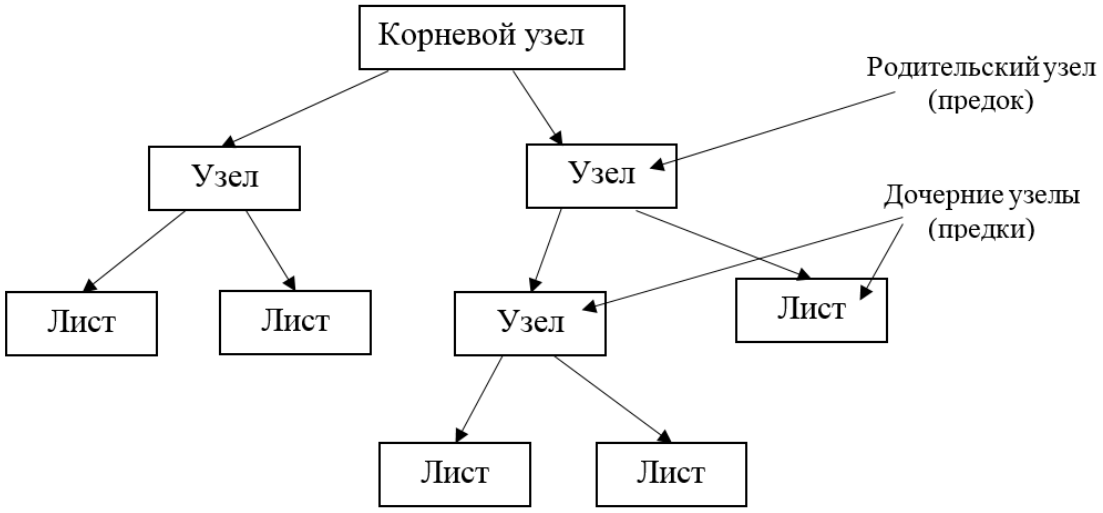

Рисунок 3.8 – Пример дерева принятия решений

Одним из способов автоматического построения деревьев решений является последовательное разбиение множества обучающих документов на классы. На каждом этапе в качестве узла дерева выбирается терм, содержащий множество всех возможных значений, и определяется условие для ветвей, затем множество документов разбивается на два класса, каждый из которых имеет свои условия [101 с. 120].

Обычно построенное дерево принятия решений является сильно детализированным (эффект переобучения с помощью адаптации модели к частным случаям, нетипичным примерам, шумам в данных и т. д.), что может привести к слишком сложным конструкциям, которые при этом недостаточно полно представляют данные. Данный алгоритм будет использован как дополнение для усовершенствования алгоритма *SVM*. А именно при помощи построения *U* уровневого дерева для каждой категории. И использовать его только для тех текстовых документов определение класса, для которых становится затруднительным, другими словами для текстов, попавших в промежуток между гиперплоскостями.

Алгоритм *SVM* с помощью обучающей выборки строит две гиперплоскости, середина между которыми и есть разделение документов на классы.

Предположим, что после построения этих гиперплоскостей появляется возможность искусственно их расширить, то есть увеличить расстояние между классами, для улучшения качества работы классификатора в соответствии с рисунком 3.9.

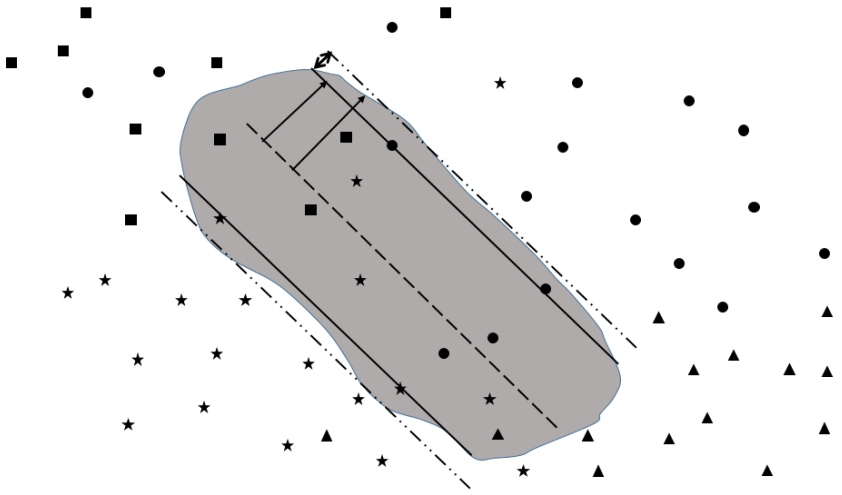

Рисунок 3.9 – Документы, имеющие сложность с определением принадлежности к тому или иному классу

Как следствие увеличится количество документов, попавших в промежуток между этими гиперплоскостями. Для определения класса принадлежности тех документов, которые попали в промежуток между гиперплоскостями, предполагается построить небольшое дерево, состоящее из некоторого количества уровней, работа которого будет осуществляться только для попавших в данный промежуток текстовых документов, для уточнения их принадлежности к тому или иному классу.

Также можно предположить, что разделяющий вектор можно увеличить. Осуществить это возможно построив его таким образом, чтобы он захватывал в неопределенное множество документы, определяющие границу вектора.

Другими словами, увеличив промежуток $\frac{2}{\|w\|}$ на k коэффициент, оптимальная величина которого будет определяться эмпирически, для всех векторов разделяющих гиперплоскости в соответствии с рисунком 3.10.

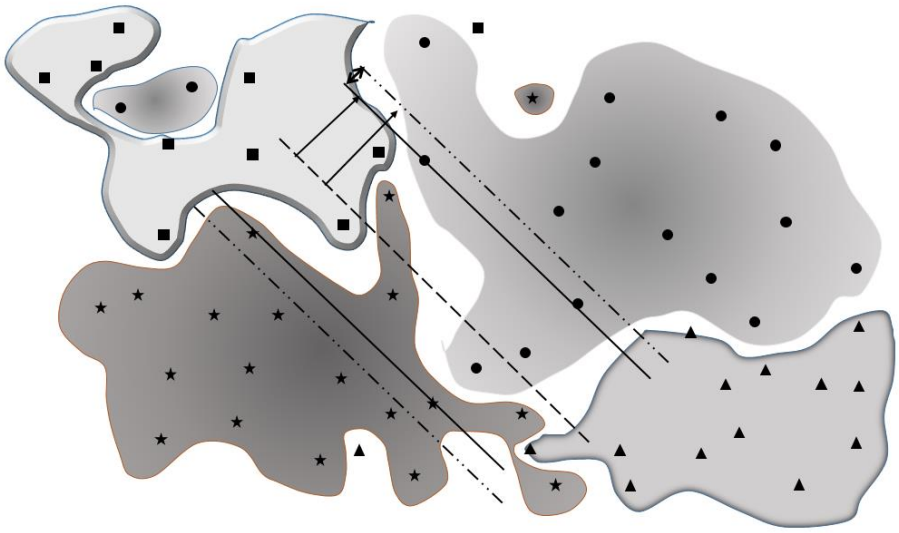

Рисунок 3.10 – Векторы разделяющие классы

Таким образом, из приведенных выше предположений, построены следующие вариации композиций методов систематизации и управления специализированной информацией, использование которых повысит качество анализа текстовых документов (Таблица 3.4).

Таблица 3.4 – Предложенные композиции методов классификации текстовых документов

Сокращение	Композиция методов	Краткое описание
1	2	3
SVM	*support vector machine*	Использование для систематизации текста одного из методов машинного обучения, а именно *support vector machine.*

Продолжение таблицы 4.3

Сокращение	Композиция методов	Краткое описание
1	2	3
SVMK	*Support vector machine* коррективы используя коэффициент *k*	Использование для систематизации текста одного из методов машинного обучения, а именно support vector machine с внесенными в него коррективами на коэффициент *k*.
SVM+NTREE	*Support vector machine* + Дерево принятия решений *U* уровней	Использование для систематизации текста одного из методов машинного обучения, а именно support vector machine и алгоритма дерево принятия решений состоящего из *U* уровней.
SVMK+NTREE	*Support vector machine* коррективы + Дерево принятия решений *U* уровней	Использование для систематизации текста одного из методов машинного обучения, а именно *support vector machine*с внесенными в него коррективами, заключающимися в увеличении границ вектора на коэффициент *k* и алгоритма дерево принятия решений состоящего из *U* уровней.

Таки образом, в таблице 3.4 представлены разработанные модернизированные композиции методов, объединившие в себе алгоритм *SVM* и дерево принятия решений, использование которых приводит к дополнительному влиянию на качество систематизации и управления информацией, повышая количество правильно распределенных по тематикам документов [56 с. 325].

Совершенствование основного алгоритма (*SVM*) при помощи дополнения его коэффициентом *k* и использованием алгоритма дерева принятия решений из *U* уровней позволяет повысить точность и полноту распределения документов по тематикам, и как следствие к повышению качества работы всей системы, обрабатывающей информацию [135].

3.6 Метрики оценки качества управления специализированной информацией

Для выявления оптимальной модели автоматической систематизации текстовой информации необходимо произвести оценку качества ее работы. Полученные оценки будут использованы для определения наиболее эффективной композиции, используемой для систематизации текстовой информации. Качество рубрицирования зависит от того, каким образом было осуществлено разбиение множества отрубрицированных документов на тестовое и обучающее множество. В данном случае стоит учитывать пару моментов:

- чем больше обучающее множество, тем лучше можно обучить алгоритм, однако, на малом тестовом множестве оценки качества могут быть слишком грубыми;
- специально подобранное разбиение отрубрицированных документов может оказать влияние на полученный итог и привести, либо к повышению, либо к понижению оценок качества [111].

Качество выстроенного классификатора оценивается при помощи его ошибки на тестовом подмножестве обучающего множества документов. Под ошибкой подразумевается доля неправильно принятых решений классификатором. Получившиеся решения классификатора сравнивают с решениями экспертов, которые формируют обучающее множество.

Статистические величины, используемые для оценки эффективности построенного классификатора вычисляются следующим образом [112 с. 4]:

Вычисление полноты $r(u)$ (*recall*) классификации информации по классам осуществляется как: «отношение количества документов, правильно приписанных к классу к общему количеству документов, относящихся к данному классу» [112 с.4]:

$$r(u) = \frac{|u \cap v|}{|v|}, \tag{2.7}$$

где

v – множество документов, принадлежащих классу,

u – множество документов, приписанных классу алгоритмом.

Точность $p(u)$ (*precision*) классификации информации по классам вычисляется как: «отношение количества документов, правильно приписанных к классу к общему количеству документов, приписанных к данному классу» [112 с. 4]:

$$p(u) = \frac{|u \cap v|}{|u|}, \tag{2.8}$$

где

v – множество документов, принадлежащих классу,

u – множество документов, приписанных классу алгоритмом.

Объединив оценки полноты и точности в одну, получим метрику качества, называемую *F*–мера (*F–measure*) [112 с. 4]:

$$F(u) = \frac{2 * p(u) * r(u)}{p(u) + r(u)}. \tag{2.9}$$

Если $p(u) = 0$ или $r(u) = 0$, то $F(u) = 0$.

Макроусреднение характеристик по всем классам вводится для получения сводных оценок качества классификации в целом.

$$Macro - p = \frac{1}{|C|} \sum_{i=1}^{|C|} p(u_i), \tag{2.10}$$

$$Macro\ -\ r\ = \frac{1}{|C|}\sum_{i=1}^{|C|} p(u_i), \quad (2.11)$$

$$Macro\ -F = \frac{1}{|C|}\sum_{i=1}^{|C|} p(u_i). \quad (2.12)$$

При условии произведения классификации документов по нескольким классам, для получения сводных оценок метрик качества используются разные методы усреднения характеристик по всем классам. Отдельная задача заключается в выборе метода усреднения. Пара наиболее часто используемых методов усреднения приведены далее: *microaverage* и *macroaverage* [136].

Допустим к каждому классу $C_1, \dots, C_n$ автоматически присвоены документы $u_1, \dots, u_n$. Следовательно, сводные оценки точности и полноты можно определить, как [4 с. 257]:

$$p_{macroavg} = \frac{1}{n}\sum_{i=1}^{n} \frac{|u_i \cap C_i|}{|u_i|}, r_{macroavg} = \frac{1}{n}\sum_{i=1}^{n} \frac{|u_i \cap C_i|}{|C_i|} \text{(макроусреднение)}, \quad (2.13)$$

$$p_{macroavg} = \sum_{i=1}^{n} \frac{|u_i \cap C_i|}{|u_i|}, r_{macroavg} = \sum_{i=1}^{n} \frac{|u_i \cap C_i|}{|C_i|} \text{ (микроусреднение)}, \quad (2.14)$$

где

C_i – количество рубрик,

u_i – документы, приписанные к этим рубрикам.

Микроусредненные и макроусредненные можно определить аналогично оценки *F*. Макроусреднение используется более часто, так как отражает поведение метода в среднем по рубрикам [4 с. 257].

Использование представленных метрик качества позволить выявить оптимальную модель, систематизирующую текстовую информацию.

3.7 Выводы к главе 3

1. Разработана обобщенная модель управления и анализа информации, способствующая улучшению качества работы систем, решающих задачи этого

класса, с целью оптимизации решения задачи управления и автоматической обработки специализированной информации. Благодаря применению разработанной модели появляется возможность оценить систему в состоянии равновесия и степень её чувствительности к различным факторам и внешним воздействиям, а также исследовать устойчивость поведения полученной модели в процессе принятия решений при обработке текстовой информации.

2. Для построения модернизированной модели управления и систематизации информации предложено использование оптимального способа анализа знаний, представляемого в виде стандарта *IDEF*. Формализация данных единым универсальным системным способом представления знаний позволяет создать соответствующие алгоритмы и подобрать инструментальные средства для обработки знаний с помощью единого формального аппарата.

3. Разработана обобщенная архитектура систематизации информации на основе предложенной модели, которая показывает общее представление ее построения и позволяет перейти к практической реализации ее прототипа и исследованию практической эффективности предложенных решений.

4. Для улучшения характеристик модели, осуществляющей систематизацию информации, предложены и обоснованы модели композиций методов систематизации, включающие в себя:

- удаление шумовых слов (позволяет уменьшить размер текстовых документов и повысит скорость работы);
- удаление окончаний, а также внесение изменений путем корректировки приоритетности слова в основном методе – *TF-IDF*, определяющий вес слова в тексте (позволяет более корректно распределить вес термов в документе, что в дальнейшем повысит качество работы классификатора);
- модернизацию метода *SVM* при помощи дополнения его алгоритмом дерево принятия решений с *U* – уровнями (позволяет повысить полноту и точность работы классификатора).

Использование предложенных композиций приводит к дополнительному влиянию на свойства предложенной модели управления и систематизации информации, и, как следствие, к повышению качества работы систем систематизации и управления текстовой информацией.

ГЛАВА 4
ПРАКТИЧЕСКАЯ РЕАЛИЗАЦИЯ И ИССЛЕДОВАНИЕ МОДИФИЦИРОВАННОЙ МОДЕЛИ СИСТЕМАТИЗАЦИИ ТЕКСТОВОЙ ИНФОРМАЦИИ

В данной главе выполнено исследование влияния предложенных композиций алгоритмов анализа на качество структуризации специализированной текстовой информации. Осуществлено исследование эффективности предложенных композиций алгоритмов для систематизации и управления информацией с использования критериев точности, полноты и *F*–меры. Использование представленных метрик качества позволит выявить оптимальную модель анализа текстовой информации для создания систем управления и систематизации текстовой информации.

4.1 Архитектура системы управления и систематизации специализированной информации средствами методологи *IDEF*

Прежде чем начинать практическую разработку системы автоматического управления и систематизации текстовой информации, необходимо синтезировать ее общую архитектуру.

При построении архитектуры системы выбор был остановлен на методологии семейства *IDEF*, использование которой эффективно отображает модули деятельности сложных систем в соответствии с рисунком 4.1 [33 с. 9].

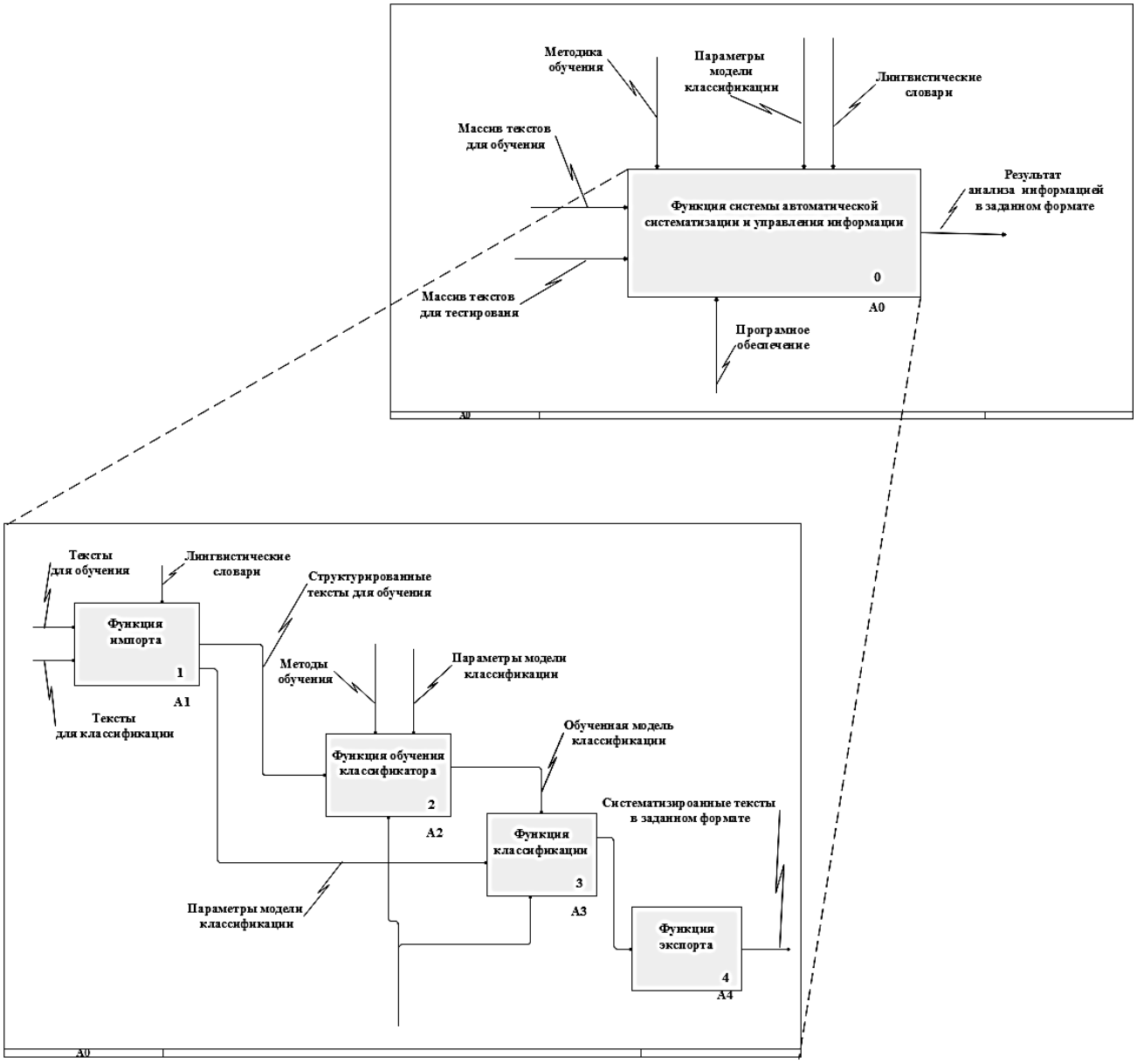

Рисунок 4.1 – Обобщенная модель системы автоматической систематизации и управления текстовой информацией

Архитектура системы, представленная в соответствии с рисунком 4.1, имеет модульную структуру, которая позволяет системе быть открытой, гибкой, и дает значительные преимущества в ее поддержке.

Принцип функциональной декомпозиции представляет собой способ моделирования ситуации управления и анализа текстовой специализированной информации, когда любое действие, операция, функция могут быть разбиты (декомпозированы) на более простые действия, операции и функции. Другими

словами, сложная организационная функция может быть представлена в виде совокупности элементарных функций, как было рассмотрено в разделе 3.2.

Прежде чем запустить работу системы, для ее корректной работы необходимо предварительно отформатировать данные. Другими словами, привести текстовые документы к единому формату. Таким образом, подготавливаются документы, подаваемые на вход для их дальнейшего отнесения системой, анализирующей информацию к той, или иной тематике. Важно заметить, что на данном этапе не происходит никакой работы с содержимым документа.

Приведя документы к единому формату, перейдем непосредственно к представлению блоков, вошедших в «Функцию автоматической систематизации и управления».

В качестве модулей выделены следующие основные функции системы:

- Функция импорта представляет собой предварительную обработку текстовых документов и используется для увеличения скорости работы алгоритма, а также приводит к дополнительному влиянию на качество систематизации и управления информацией.
- Функция обучения классификатора предполагает выполнение корректировки рубрик для повышения качества классификации текстовой информации.
- Функция классификации, объединяет в себе несколько алгоритмов, а именно *SVM* и дерево принятия решений, использование которых приводит к дополнительному влиянию на качество систематизации и управления информацией.
- Функция экспорта, экспортирует документы в определённую системой управляющей и систематизирующей информацию тематику.

Анализируя результаты опубликованных экспериментов [119 с. 30; 124 с. 97], сравнение которых подробно рассмотрено в разделе 2.6, можно сделать вывод о том, что конечный результат качества отнесения документов по темникам

зависит от предобработки не в меньшей степени, чем от метода машинного обучения.

В данной работе были представлены различные варианты дальнейшей модернизации предложенной модели управления и систематизации информации, в которой недостатки отдельных классических алгоритмов взаимно компенсируются. Для выбора наиболее эффективной модели систематизации информации для предсказания тематики или набора тематик релевантных для нового текстового документа, использованы метрики качества.

Критерии качества (полнота, точность, *F*–мера) были рассчитаны по итогам работы предложенной модели управления и систематизации текстовой информации для различных вариаций композиций моделей систематизации информации, описание которых представлено в главе 3.

4.2 Требования к исходным данным и исследование предварительной обработки корпуса текстов

В сфере компьютерных технологий есть множество вариантов представления данных, которые используются для преобразования их в рамках компьютерной среды. Преобразование может быть прямым, как в случае модернизации до более новой версии компьютерной программы. В альтернативном варианте конвертация может потребовать использование специальной конвертирующей программы, а также может включать сложный процесс прохождения промежуточных стадий или вовлечения сложных «экспортирующих» и «импортирующих» процессов перехода от одного формата к другому в соответствии с рисунком 4.2.

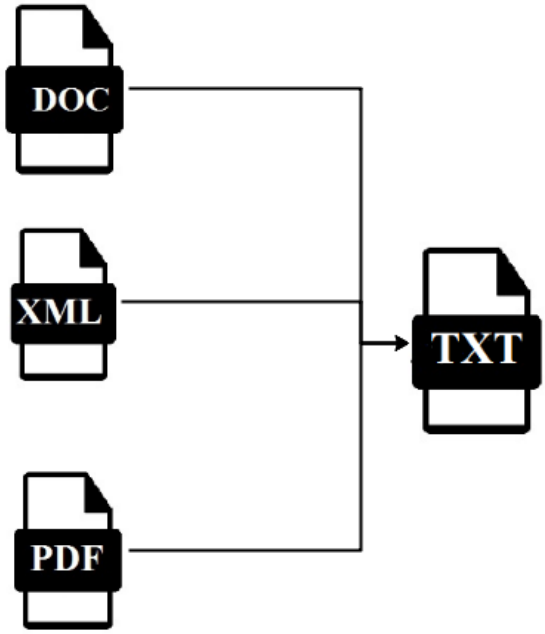

Рисунок 4.2 – Приведение файлов к единому формату

Для осуществления конвертации файлов, на сегодняшний день, существует масса готовых сервисов, предлагающих решение данного вопроса. Существует множество возможных сценариев. Можно использовать как готовый алгоритм, так и написать собственный.

В данной работе в качестве текстов для проведения исследований используются русскоязычные текстовые документы в электронном виде.

Средняя длина текстового документа составляет от нескольких предложений до нескольких десятков страниц, включая таблицы и рисунки [137].

В исследование участвует два корпуса документов, на одном из которых осуществляется тестирование, а на другом обучение. Объем данных составляет 5000 текстов, для каждого из корпусов. Все исследования проводились самостоятельно, так как в зависимости от набора текстовых документов меняется и качество распределения текстов по тематикам, что приведет к ошибкам в показателях полноты, точности и *F*–меры. Для сравнения с модифицированными моделями композиций, использован существующей метод *SVM*, тестирование которого осуществляется на подобранном объёме документов.

Исходя из личного опыта на обработку одного текста, среднего размера, затрачивается примерно 10 – 30 минут. Таким образом, на обработку 5000 текстов в среднем необходимо затратить от месяца до трех круглосуточной работы. Как следствие на момент получения необходимой информации появляется вероятность потери ее актуальности (по различным причинам), что обуславливает необходимость модернизации и реализации системы, автоматически обрабатывающей текстовую информацию.

Одним из этапов предварительной обработки текстовых документов является выбор характеристик, которые влияют на качество работы исследуемой модели системы управления и систематизации информации. Для исследования использованы предложенные в таблице 3.1 композиции методов, уменьшающие размер текста. Обработка текстовой информации, а именно удаление неинформативных слов и окончаний, представлено в таблице 4.1.

Приведенный пример предварительной обработки текста был осуществлен на предложениях, случайно изъятых из одного из текстовых документов, вошедших в тестовую выборку [138].

Таблица 4.1 – Пример предварительной обработки текстовых данных

Исходный текст	Обработанный текст
1	2
Цифровые средства, используемые для создания CG NPR–иллюстраций, – графические редакторы Adobe Photoshop, Painter, аппаратные средства – графические планшеты дифференцируемых конфигураций разных производителей.	Цифров средст используем создан CG NPR иллюстрац графическ редактор Adobe Photoshop Painter аппарат средст графич планшет дифференцируем конфигурац разн производите
Основная задача NPR – совместить красоту и эмоциональность традиционной художественной изобразительности со скоростью и гибкостью компьютерной графики.	Основ задач NPR совмест красот эмоционал традици худож изобразительн скорост гибкост компьютер график.

Продолжение таблицы 4.1

Исходный текст	Обработанный текст
1	2
При этом большая часть методов NPR моделирует традиционные художественные формы: 1) живопись: акварель, методы, имитирующие мазки краски различной конфигурации; 2) рисование: чернила, карандаш, уголь, пастель.	больш част метод NPR модел традиц худож формы живопис акварел метод имитирующ мазк краск различ конфигурац рисован чернил карандаш угол пастел

Таким образом, использование стемминга и удаление стоп слов предоставляют возможность осуществить уменьшение размера текстового документа, с минимальными потерями смысловой нагрузки текста. Обработка при помощи стемминга позволит понять электронной вычислительной машине, что то или иное слово встречается не шесть раз, а двадцать один раз, разница существенна. Например, слово «цвет» встречается в данном текстовом документе в таких вариациях: цвет, цветом, цвета. Эксперт может понять, что это одно и тоже слово, а вычислительная машина нет, так как она будет сравнивать слова посимвольно.

Для работы классификатора необходимы тексты, преобразованные в векторный вид. Одним из самых распространенных методов построения вектора, определяющего вес термов, является метод *TF–IDF* [56 с. 328].

В качестве примера приведем сравнение значимости слова для различных композиций, представленных в разделе 3 таблице 3.1 по весу.

Для наглядности, в таблице 4.2 приведен вес сорока наиболее часто встречающихся слов в документе. Нулем в таблице обозначены слова, отсутствующие и не принимающие участия в построении вектора используемого классификатором.

Таблица 4.2 – Значения *TF–IDF* с использованием композиций.

Слова	Композиции			
	Т–I	Ш+Т–I	С+Т–I	С+Ш+Т–I
1	2	3	4	5
цветом	0,03311	0,03311	0,08277	0,08277
для	0,03834	0	0,03834	0
компьютерной	0,03661	0,03661	0,09885	0,09885
или	0,04297	0	0,04726	0
графики	0,03055	0,03055	0,10861	0,10861
иллюстрации	0,02604	0,02604	0,08463	0,08463
что	0,02219	0	0,02773	0
при	0,02917	0	0,02917	0
слой	0,03179	0,03179	0,09935	0,09935
как	0,02181	0	0,02493	0
изображения	0,03337	0,03337	0,13349	0,13349
графика	0,03043	0,03043	0	0
вид	0,01687	0	0,03092	0,03092
the	0,02546	0	0,0297	0
это	0,02167	0	0,0578	0
изображение	0,03043	0,03043	0	0
текста	0,02546	0,02546	0,05091	0,05091
кисти	0,01758	0,01758	0,03223	0,03223
цвет	0,02323	0,02323	0	0
book	0,02367	0,02367	0,02367	0,02367
искусства	0,01958	0,01958	0,03133	0,03133
смысле	0,0161	0,0161	0,0161	0,0161
его	0,01441	0	0,01441	0
слоев	0,01676	0,01676	0	0
компьютерная	0,01085	0,01085	0	0
книжной	0,01633	0,01633	0,08164	0,08164
позволяет	0,01851	0,01851	0,0324	0,0324
иллюстрация	0,01973	0,01973	0,08879	0,08879

Продолжение таблицы 4.2

Слова	Композиции			
	Т–I	Ш+Т–I	С+Т–I	С+Ш+Т–I
1	2	3	4	5
graphics	0,02006	0,02006	0,02006	0,02006
компьютерных	0,01465	0,01465	0	0
проблемы	0,01613	0,01613	0,0242	0,0242
данной	0,01811	0,01811	0,04075	0,04075
этом	0,01432	0	0,03579	0
методы	0,01966	0,01966	0,02949	0,02949
этот	0,01091	0	0	0
процесс	0,01136	0,01136	0,02272	0,02272
наброска	0,01096	0,01096	0,02192	0,02192
создается	0,01826	0,01826	0,05479	0,05479
заполнения	0,01836	0,01836	0,03212	0,03212
слоя	0,01618	0,01618	0	0

В зависимости от того, какая композиция будет использована, будет меняться и длина вектора, а также и значимость, то есть вес слова в документе таблица 4.2.

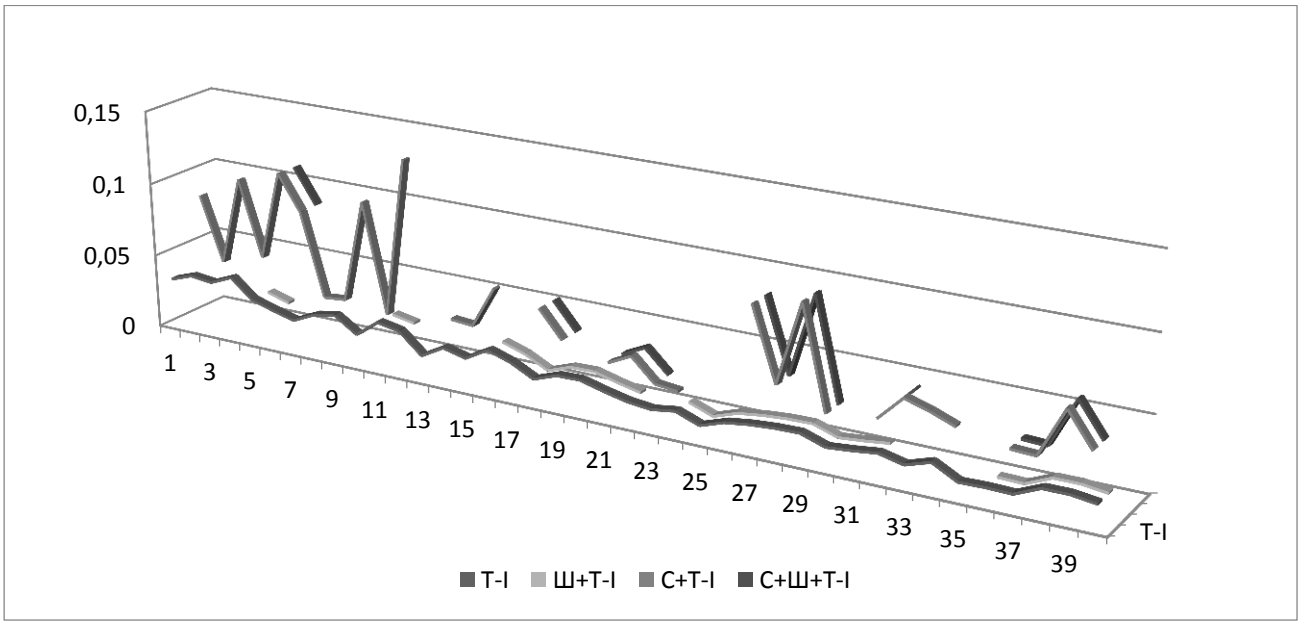

Рисунок 4.3 – Изменение длины вектора

В соответствии с рисунком 4.3 изображены точки вектора, получившиеся при обработке различными композициями каждый вектор для одного и того же документа, имеет разное количество слов, имеющих свой вес, варьирующийся в зависимости от используемой композиции. Уменьшая количество слов, другими словами количество точек вектора, обеспечим большее быстродействие алгоритма [139].

Необходимо учитывать факт того, что чрезмерное уменьшение вектора или ограничение в доступе к тексту приводит к затруднению в выявлении важных слов в документе, при помощи которых будет определяться его тематика и отнесение к тому или иному классу.

Таким образом, возникает необходимость в проведении дополнительных экспериментов, выявляющих уровень качества определения тематики документа при использовании представленных в таблице 4.2 композиций классификации.

Основой для проведения экспериментов стал алгоритм *support vector machine* [4 с. 131], а результаты тестирования приведены в таблице 4.3.

Таблица 4.3 – Метрики качества методов уменьшения размерности вектора

Название композиций	Точность,%	Полнота,%	F–мера, %
1	2	4	5
Т–I	66	68	67
Ш+Т–I	75	73	74
С+Т–I	68	59	63
С+Ш+Т–I	86	84	85

Основываясь на результатах проведенных экспериментов, представленных в таблице 4.3, можно сделать вывод о том, что на данном этапе оптимальным решением будет использование композиции, состоящей из сочетания «Стемминг» + «Стоп–слова» + *TF–IDF*. Эта композиция позволит облегчить работу метода *TF–IDF*, избавив его от неинформативных слов и преобразовав слова к общей форме.

Следующим этапом исследования данного блока будет изучение влияния дополнительных коэффициентов на качество классификации текстового документа. Для определения метрик качества за основу взят все тот же метод *SVM* [140].

В таблицах 4.4 и 4.5 рассмотрены изменения веса слов, с учетом различных вариаций коэффициента *L*, при использовании композиций, представленных в таблице 3.2 раздела 3.3. За основу, в данных экспериментах, взят блок, состоящий из сочетания «Стемминг» + «Стоп–слова» + *TF–IDF* с наилучшими показателями качества. В результате применения композиции, состоящей из удаления шумовых слов и стеммов, вектор уменьшается практически в два раза. Это позволяет ускорить работу алгоритма, избавив его от ненужных вычислений, а также повысить качество определения веса для каждого текстового документа.

Таблица 4.4 – Композиции векторов увеличения веса терма

Слова	Композиции			
	С+Ш+Т–I	L+С+Ш+Т–I	L+Ш+Т–I –2	L+ С+Ш+Т–I–3
1	2	3	4	5
цветом	0,08095	0,08286	0,00254	0,11345
компьютерной	0,024285	0,09899	0,00362	0,13549
графики	0,045446	0,10875	0,00437	0,14886
иллюстрации	0,058464	0,08464	0,00044	0,08983
слой	0,021081	0,0995	0,00366	0,13617
изображения	0,017171	0,13372	0,0066	0,18296
вид	0,008994	0,03095	0,00035	0,04238
текста	0,008521	0,05099	0,00096	0,06978
кисти	0,01746	0,03226	0,00039	0,04417
book	0,012265	0,02371	0,00021	0,03244
искусства	0,016605	0,03138	0,00036	0,04294
смысле	0,006425	0,01612	9,61E–05	0,02207
книжной	0,038548	0,08177	0,00247	0,1119
позволяет	0,00787	0,03245	0,00039	0,04441

Продолжение таблицы 4.4

Слова	Композиции			
	С+Ш+Т–I	L+С+Ш+Т–I	L+Ш+Т–I –2	L+ С+Ш+Т–I–3
1	2	3	4	5
иллюстрация	0,014285	0,08896	0,00292	0,1217
graphics	0,013492	0,0201	0,00015	0,0275
проблемы	0,013492	0,02424	0,00022	0,03317
данной	0,010559	0,04081	0,00062	0,05585
методы	0,004626	0,02954	0,00032	0,04042
процесс	0,009812	0,02275	0,00019	0,03114
наброска	0,019624	0,02194	0,00018	0,03004
создается	0,023129	0,05488	0,00111	0,07509
заполнения	0,008586	0,03218	0,00038	0,04403

Таблица 4.5 – Композиции векторов увеличения веса терма

Слова	Композиции		
	С+Ш+Т–I	L+ С+Ш+Т–I–4	L+ С+Ш+Т–I –5
1	2	3	3
цветом	0,08095	0,002539	0,03321
компьютерной	0,024285	0,003621	0,04026
графики	0,045446	0,004372	0,04462
иллюстрации	0,058464	0,00044	0,00564
слой	0,021081	0,003658	0,04048
изображения	0,017171	0,006604	0,05607
вид	0,008994	0,000354	0,01181
текста	0,008521	0,000961	0,01983
кисти	0,01746	0,000385	0,01233
book	0,012265	0,000208	0,00898
искусства	0,016605	0,000364	0,01198
смысле	0,006425	9,61E–05	0,00606
книжной	0,038548	0,00247	0,03273
позволяет	0,00787	0,000389	0,0124

Продолжение таблицы 4.5

Слова	Композиции		
	С+Ш+Т–I	L+ С+Ш+Т–I–4	L+ С+Ш+Т–I –5
1	2	3	3
Иллюстрация	0,014285	0,002922	0,03583
graphics	0,013492	0,000149	0,00758
проблемы	0,013492	0,000217	0,00919
данной	0,010559	0,000615	0,01572
методы	0,004626	0,000322	0,01125
процесс	0,009812	0,000191	0,00861
наброска	0,019624	0,000178	0,0083
создается	0,023129	0,001112	0,02142
заполнения	0,008586	0,000382	0,01229

В зависимости от того, какая композиция будет использована, будет меняться и вес терма в соответствии с рисунком 4.4 в непосредственно взятом векторе, а как следствие и значимость, то есть вес терма, как в документе, так и в корпусе текстов таблицы 4.4 и 4.5.

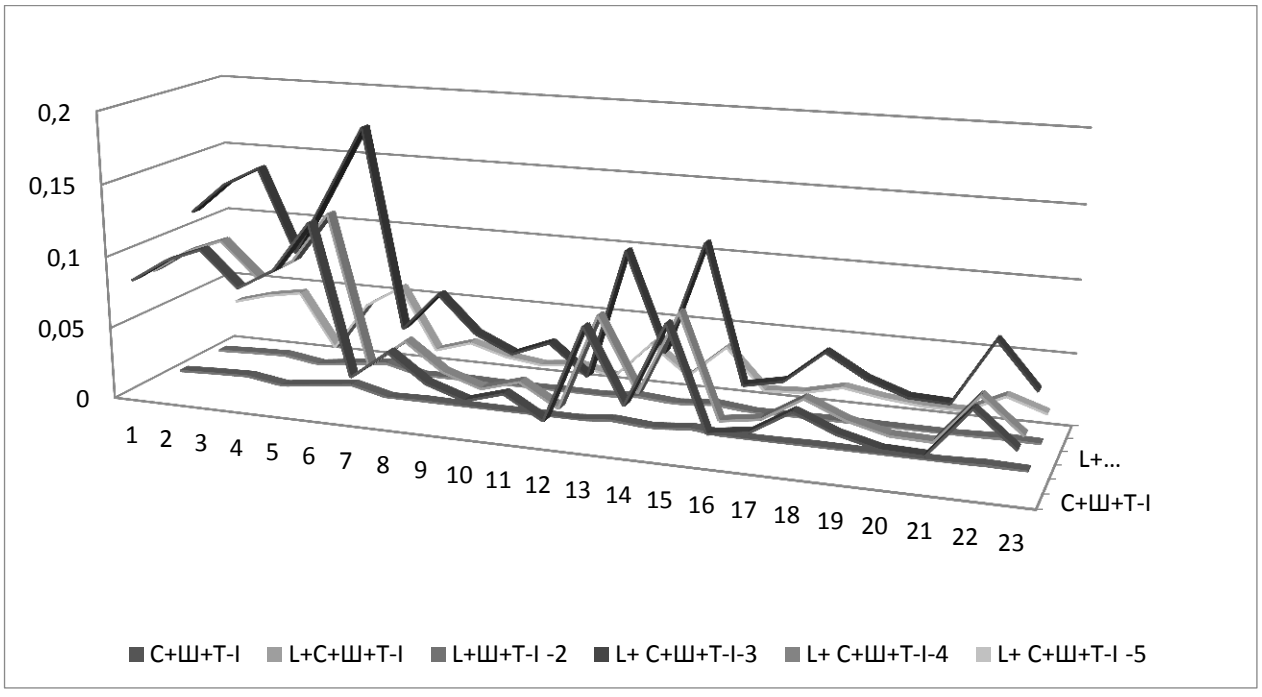

Рисунок 4.4 – Вариации веса с использованием коэффициента *L*

В соответствии с рисунком 4.4 изображены точки вектора, получившиеся при обработке композициями, включающими в себя коэффициент *L*. Для получения более высоких результатов незначительно увеличивается время работы классификатора.

В таблице 4.6 представлено то, каким образом повлияет использование коэффициента *L*, при различных композициях, на качество классификации. Для определения качества за основу принимается метод классификации *SVM*.

Таблица 4.6 – Метрики качества определения веса терма.

Название композиций	Точность,%	Полнота,%	F– мера
1	2	3	4
(Т–I)+С+Ш	86	84	85
((Т–I)+С+Ш)+L–1	87	83	85
((Т–I)+С+Ш)+L–2	89	91	90
((Т–I)+С+Ш)+L–3	95	92	93,5
((Т–I)+С+Ш)+L–4	89	93	90

В результате проведения серии экспериментов и основываясь на метриках качества, приведённых в таблице 4.6, сделан вывод о том, что оптимальным будет третье сочетание, состоящее из «Доп. коэффициент *L*» + $TF - IDF(w, d, D)$ +«Стемминг» + «Шумовые слова», что можно рассчитать по формуле:

$$\mathrm{TF} - \mathrm{IDF}(\mathrm{w}, \mathrm{d}, \mathrm{D}) = \frac{\mathrm{n_i}}{\sum_k \mathrm{n_k}} \times \frac{|\mathrm{D}|}{|(\mathrm{d_i} \supset \mathrm{w_i})|} + \mathrm{L} + \mathrm{C} + \mathrm{Ш} =$$

$$= \frac{\mathrm{n_i}}{\sum_k \mathrm{n_k}} \times \frac{|\mathrm{D}|}{|(\mathrm{d_i} \supset \mathrm{w_i})|} + \frac{\mathrm{n_i}}{|(\mathrm{d_i} \supset \mathrm{w_i})|} + \mathrm{C} + \mathrm{Ш}. \quad (4.1)$$

Эта композиция позволит улучшить работу метода *TF–IDF*, изменив распределение веса термов в текстовых документах, что повысит качество работы классификатора.

Завершающим этапом исследования данного блока будет изучение влияния дополнительных коэффициентов *Q* и *Z*, на качество распределения веса в текстовом документе.

За основу для определения метрик качества будет использован метод *SVM* [95 с. 55].

Для начала рассмотрим изменение веса слов при использовании композиций, представленных в таблице 3.2 раздела 3.3. Для проведения дальнейшего исследования взята композиция, показавшая наилучшие результаты на предыдущем этапе тестирования.

В таблицах 4.7 и 4.8 представлены результаты распределения веса слов с учетом преобразования формулы, путем добавления в нее коэффициентов *Q* и *Z*. Для построения вектора применяются составленные композиции, использование которых способствуют повышению качества обработки текстовых документов.

Таблица 4.7 – Вес термов с учетом коэффициентов *Q* и *Z*

Слова	Композиции			
	L+Стем+ Стоп+TFIDF–3	(L+Стем+ Стоп+TFIDF–3) *+Q+Z–1*	(L+Стем+ Стоп+TFIDF–3)*+Q+Z–2*	(L+Стем+ Стоп+TFIDF–3)*+Q–1*
1	2	3	4	5
1. цвет	0,113446	0,114825	0,613446	0,114825
3. компьютер	0,135488	0,135488	1,135488	0,137319
5. граф	0,148863	0,148863	1,148863	0,150561
6. иллюстрац	0,089828	0,089828	0,089828	0,089828
9. сло	0,136168	0,136168	0,136168	0,136168
11. изображен	0,182956	0,182956	0,682956	0,18534
13. вид	0,042377	0,042377	0,042377	0,042377
17. текст	0,06978	0,06978	0,06978	0,06978
18. кист	0,044171	0,044171	0,544171	0,045636
20. book	0,032441	0,032441	0,032441	0,032441
21. искусства	0,042941	0,042941	0,042941	0,042941
22. смысл	0,022066	0.022066	0,022066	0,022066

Продолжение таблицы 4.7

Слова	Композиции			
	L+Стем+ Стоп+TFIDF–3	(L+Стем+ Стоп+TFIDF–3) +*Q*+*Z–1*	(L+Стем+ Стоп+TFIDF–3)+*Q*+*Z–2*	(L+Стем+ Стоп+TFIDF–3*)*+*Q–1*
1	2	3	4	5
26. кни	0,111901	0,111901	0,111901	0,111901
27. позволяет	0,044405	0,044405	0,044405	0,044405
28. иллюстрация	0,1217	0,1217	0,1217	0,1217
29. graphics	0,027497	0,027497	1,027497	0,030005
31. проблемы	0,033169	0,033169	0,033169	0,033169
32. данной	0,055847	0,055847	0,055847	0,055847
34. методы	0,040419	0,040419	0,040419	0,040419
36. процесс	0,031144	0,031144	0,031144	0,031144
37. наброска	0,030037	0,030037	0,030037	0,030037
38. создается	0,075093	0,075093	0,075093	0,075093
39. заполнения	0,044028	0,044028	0,044028	0,044028

Таблица 4.8 – Вес термов с учетом коэффициентов *Q* и *Z*

Слова	Композиции			
	L+Стем+ Стоп+TFIDF–3	Z+L+Стем+ Стоп+TFIDF–2	Q+L+Стем+ Стоп+TFIDF–1	Q+L+Стем+ Стоп+TFIDF–2
1	2	3	4	5
1. цвет	0,113446	0,613446	0,113446	0,113446
3. компьютер	0,135488	0,635488	0,137319	0,635488
5. граф	0,148863	0,648863	0,150561	0,648863
6. иллюстрац	0,089828	0,089828	0,089828	0,089828
9. сло	0,136168	0,136168	0,136168	0,136168
11. изображен	0,182956	0,682956	0,182956	0,182956
13. вид	0,042377	0,042377	0,042377	0,042377
17. текст	0,06978	0,06978	0,06978	0,06978
18. кист	0,044171	0,544171	0,044171	0,044171
20. book	0,032441	0,032441	0,032441	0,032441

Продолжение таблицы 4.8

Слова	Композиции			
	L+Стем+ Стоп+TFIDF–3	Z+L+Стем+ Стоп+TFIDF–2	Q+L+Стем+ Стоп+TFIDF–1	Q+L+Стем+ Стоп+TFIDF–2
1	2	3	4	5
21. искусства	0,042941	0,042941	0,042941	0,042941
22. смысл	0,022066	0,022066	0,022066	0,022066
26. кни	0,111901	0,111901	0,111901	0,111901
27. позволяет	0,044405	0,044405	0,044405	0,044405
28. иллюстрация	0,1217	0,1217	0,1217	0,1217
29. graphics	0,027497	0,527497	0,030005	0,527497
31. проблемы	0,033169	0,033169	0,033169	0,033169
32. данной	0,055847	0,055847	0,055847	0,055847
34. методы	0,040419	0,040419	0,040419	0,040419
36. процесс	0,031144	0,031144	0,031144	0,031144
37. наброска	0,030037	0,030037	0,030037	0,030037
38. создается	0,075093	0,075093	0,075093	0,075093
39. заполнения	0,044028	0,044028	0,044028	0,044028

В зависимости от того, какая композиция будет использована, будет меняться и вес терма в непосредственно взятом векторе, а как следствие и значимость, то есть вес терма, как в документе, так и в корпусе текстов (Таблицы 4.7 и 4.8).

В соответствии с рисунком 4.5 наглядно представлено изменение веса терма в зависимости от того при помощи какой композиции он будет определяться.

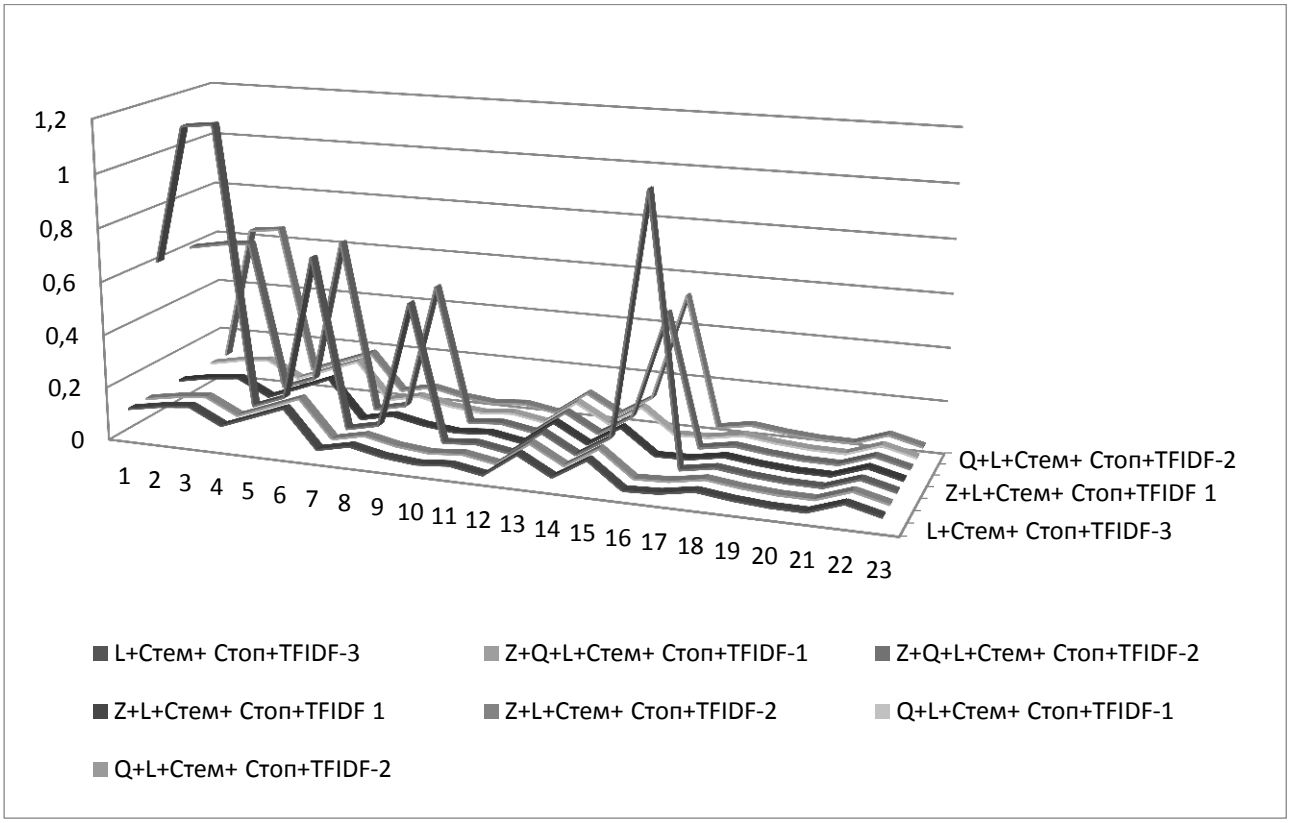

Рисунок 4.5 **–** Вариации веса с использованием коэффициента *Z* и *Q*

Опираясь на показатели использования метрик качества (Таблица 4.9) получены наилучшие результаты работы составленных композиций методов Z+Q+L+Стем+Стоп+TF–IDF–2.

Таблица 4.9 – Метрики качества определения веса терма

Название композиций	Точность,%	Полнота,%	F– мера
1	2	3	4
((Т–І)+С+Ш)+L–3	95	92	93,5
(((Т–І)+С+Ш)+L–3)+Q+Z–1	94	93	93,5
(((Т–І)+С+Ш)+L–3)+Q+Z–2	96	94	95
(((Т–І)+С+Ш)+L–3)+Q–1	95	92	94
(((Т–І)+С+Ш)+L–3)+Q–1	93	89	91
(((Т–І)+С+Ш)+L–3)+Z–1	94	92	93
(((Т–І)+С+Ш)+L–3)+Z–2	94	96	95

Вследствие проведенных экспериментов оптимальной композицией векторизации текстовой информации является следующая композиция, описываемая моделью:

$$(((T–I)+C+Ш\ )+L–3)+Q+Z–2. \quad (4.2)$$

При построении «функции импорта» данная композиция получила наилучшие показатели метрик качества, объединив в себе несколько методик обеспечивающих предварительную обработку информации, построенная композиция включает в себя следующие шаги:

- удаление шумовых слов уменьшает размер текстовых документов и повышает скорость работы алгоритма;
- удаление окончаний и использование коэффициентов *L, Q, Z* позволяет корректно распределить вес термов в документе, что повышает качество работы классификатора.

Полученная модернизированная композиция построения вектора будет использована в следующем модуле для определения тематики текстовой информации.

4.3 Исследование модуля классификации

В результате проведенных исследований получены композиции методов, объединившие в себе положительные качества нескольких композиций, описание которых приведено в разделе 3.5. Данные композиции объединяют в себе методы машинного обучения, а именно [56 с. 329]:

- алгоритм *support vector machine*;
- алгоритм дерево принятия решений.

Использование данных алгоритмов приведёт к дополнительному влиянию на качество систематизации информации. В результате проведенных исследований получены вариации композиций методов обработки текстовой

информации, представленные в таблице 3.4 (приведены дополняющие друг друга комбинации методов, используемые для повышения качества обработки текстовых документов) [56 с. 329].

Данные методы совместно образуют композиции, результаты качества которых представлены в таблице 4.10.

Таблица 4.10 – Метрики качества моделей классификаторов

Название композиций	Точность, %	Полнота, %	F– мера, %
1	2	3	4
(((Т–І)+С+Ш)+L–3)+Q+Z–2 +SVM	96	94	95
(((Т–І)+С+Ш)+L–3)+Q+Z–2 +SVMK	90	94	92,5
(((Т–І)+С+Ш)+L–3)+Q+Z–2 +SVM +NTREE	98,5	99,5	99
(((Т–І)+С+Ш)+L–3)+Q+Z–2 SVMK +NTREE	98	94	96

На основании данных, представленных в таблице 4.10, можно сделать вывод, что наилучшие показатели, при обработке русскоязычных текстовых документов, получила модель:

$$(((Т–І)+С+Ш)+L–3)+Q+Z–2 +SVM +NTREE. \quad (4.3)$$

Данная модель объединила в себе синтагматический и парадигматический подходы. Так как использование разработанной модели входит в процесс построения модернизированной системы управления и систематизации информации, то качество распределения информации соответствует 99%.

Данная модель позволила повысить качество распределения документов по тематикам. Это в свою очередь позволит своевременно получить доступ к актуальной информации и ускорить анализ необходимых сведений.

4.4 Обобщенная оценка практической эффективности разработок

Применение обобщенной модели, распределяющей данные по тематикам, предоставляет возможность оценить систему управления и систематизации информации в состоянии равновесия и степень её чувствительности к различным внешним воздействиям, а также исследовать устойчивость поведения полученной структуры, способствующей управлению и систематизации текстовой информации.

При создании улучшенной основы представления знаний и управления ими, появляется возможность отследить влияние того или иного воздействия на поведение системы управления и систематизации информации. Так в таблице 4.11 предоставлено качество работы различных модернизированных композиций моделей управления и систематизации информации, объединяющих в себе некоторые методики, дополняющие работу друг друга с цель получения наилучшего качества при распределении данных по тематикам.

Таблица 4.11 – Критерии качества различных композиций моделей специализированной информации

Название композиций	Точность,%	Полнота,%	F– мера, %
1	2	3	4
Т–I	66	68	67
Ш+Т–I	75	73	74
С+Т–I	68	59	63
С+Ш+Т–I	86	84	85
(Т–I)+С+Ш	86	84	85
((Т–I)+С+Ш)+L–1	87	83	85
((Т–I)+С+Ш)+L–2	89	91	90
((Т–I)+С+Ш)+L–3	95	92	93,50
((Т–I)+С+Ш)+L–4	89	93	90
(((Т–I)+С+Ш)+L–3)+Q+Z–1	94	93	93,50

Продолжение таблицы 4.11

Название композиций	Точность,%	Полнота,%	F– мера, %
1	2	3	4
(((Т–І)+С+Ш)+L–3)+Q+Z–2	96	94	95
(((Т–І)+С+Ш)+L–3)+Q–1	95	92	94
(((Т–І)+С+Ш)+L–3)+Q–2	93	89	91
(((Т–І)+С+Ш)+L–3)+Z–1	94	92	93
(((Т–І)+С+Ш)+L–3)+Z–2	94	96	94,5
(((Т–І)+С+Ш)+L–3)+Q+Z–2 +SVM	96	94	95
(((Т–І)+С+Ш)+L–3)+Q+Z–2 +SVMK	90	94	92,50
(((Т–І)+С+Ш)+L–3)+Q+Z–2 +SVM+NTREE	98,50	99,50	99
(((Т–І)+С+Ш)+L–3)+Q+Z–2SVMK+NTREE	98	94	96

В соответствии с рисунком 4.6 наглядно продемонстрированы результаты экспериментальных исследований, направленных на сравнение эффективности используемых способов модернизации моделей управления и систематизации информации.

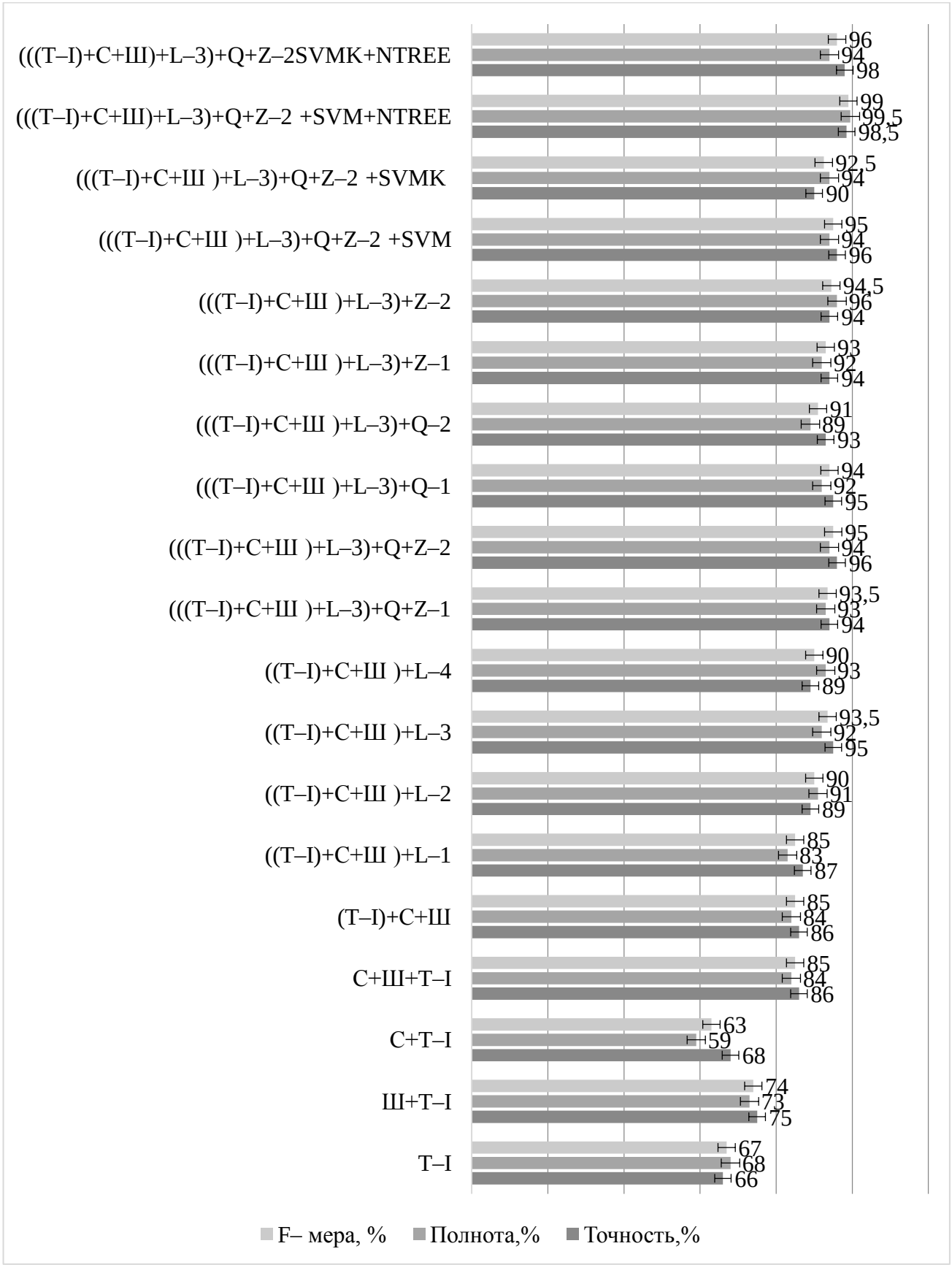

Рисунок 4.6 – Итоговая диаграмма качества композиций, обрабатывающих информацию

Подводя итог исследования модернизированной композиции, объединяющей в себе несколько методик, наилучшие показатели качества систематизации информации продемонстрировала модель:

$$(((\text{Т–I})+\text{C}+\text{Ш})+\text{L–3})+\text{Q}+\text{Z–2}+\text{SVM}+\text{NTREE}, \quad (4.4)$$

где

Т–I – использование метода $TF - IDF$,

С – удаление стемминга,

Ш – удаление шумовых слов,

L–3 – при распределении веса учитывает коэффициент L,

Q+Z–2 – при распределении веса учитывает коэффициенты Q и Z,

SVM+NTREE – объединяет метод *support vector machine* и алгоритм дерево принятия решений состоящий из U уровней.

Разработана обобщенная архитектура системы систематизации информации на основе предложенной модели (4.4), которая демонстрирует критерии качества, отображающие 99% правильно отнесенных текстовых документов по тематикам.

Таким образом, в предложенную модернизированную модель системы управления и систематизации информации, были включены следующие модификации, улучшившие ее работу:

- удаление шумовых слов позволило уменьшить размер текстовых документов и повысит скорость работы алгоритма;
- удаление окончаний и использование коэффициентов L, Q, Z позволило корректно распределить вес термов в документе, что повысило качество работы классификатора;
- модернизация алгоритма SVM при помощи дополнения его путем использования алгоритма дерево принятия решений, состоящего из U уровней, повысило полноту и точность работы классификатора.

Использование предложенных модифицирующих свойств, привело к повышению качества работы модели (4.4) систематизирующей текстовые

документы, и как следствие повысило качество работы всей системы управления специализированной информацией на 5%.

4.5 Выводы к главе 4

В главе представлено практическое исследование эффективности разработанных средств, при использовании тематического анализа коллекции документов. Осуществлено практическое сравнение методов машинного обучения при использовании различных средств классификации и предложены эффективные модели модернизированных композиций, объединяющих в себе несколько методик, для получения наилучших показателей качества систематизации информации.

1. Экспериментальное исследование предложенных средств показало их позитивное влияние на усовершенствование предложенной общей модели систематизации и управления информацией на основе правил отбора неинформативных признаков, способов взвешивания термов и композиций методов систематизации.

2. Экспериментальное исследование моделей вычислительных композиций распределения веса слов в текстовом документе показало, что при разном распределении веса терма повышение качества систематизации информации варьируется в пределах 10% в зависимости от используемой композиции.

3. Экспериментальное исследование модели классификации текстовой информации на основе предложенных композиций методов систематизации и внесения изменений в структуру алгоритма её построения показало, что полнота и точность работы модели повышается на 5,5% и 8,5 % соответственно.

4. Экспериментальное исследование предложенной общей модели автоматической систематизации и управления информацией, основанной на выполненном объединении достоинств синтагматических и парадигматических

подходов, и всех предложенных модернизаций показало увеличение полноты и точности работы в среднем на 32,5% и 31,5% соответственно.

5. Экспериментальное исследование предложенной общей модернизированной модели управления и систематизации текстовых документов на основе всех разработанных средств показало повышение скорости и качества обработки текстовой информации в среднем более чем в 4 раза по сравнению с ручным способом (так на обработку одного текстового документа вручную в среднем затрачивается 10 - 30 минут, в то время как автоматическая обработка на основе разработанных средств – в среднем затрачивает 3 - 7 минут).

6. Результаты исследований имеют широкий спектр применения для различных предметных областей. Предложенная практическая реализация усовершенствованной модели управления и систематизации информации позволяет формировать текстовые базы данных, содержащие классифицированную информацию, в автоматическом режиме. На основании результатов классификации появляется возможность автоматизировать работу специалистов-аналитиков, осуществляющих тематический анализ текстовой информации, и ведение аналитических задач в различных предметных областях, что может послужить функциональным дополнением и развитием информационных систем различных организаций.

ЗАКЛЮЧЕНИЕ

Диссертация является законченной научно-исследовательской работой, в которой получено решение важной научно-технической задачи повышения эффективности управления и систематизации специализированной текстовой информации больших объёмов с учетом особенностей русскоязычных текстовых документов, реализованных с помощью модернизация моделей, методик, и алгоритмов системного анализа, искусственного интеллекта и компьютерной лингвистики.

Основные научные результаты и выводы, полученные при выполнении работы, состоят в следующем:

1. Управление и систематизация по содержанию большого количества текстов является сложной актуальной задачей. Ее выполнение ограниченным количеством специалистов и затрачиваемым временем в условиях постоянного поступления новой информации практически невозможно.

2. Анализ современных публикаций позволяет утверждать, что существует значительный разрыв между методами систематизации и управления информацией, основанными на машинном обучении, и методами основанными на знаниях.

3. Выполнен анализ принципов построения систем управления, систематизации и обработки текстовой информации, изучены особенности их работы. Формализация единого универсального системного способа представления знаний позволяет создать соответствующие алгоритмы и инструментальные средства для обработки знаний различного типа единообразным способом и с помощью единого формального аппарата, построение которого осуществляется на основе методов и алгоритмов машинного обучения в рамках анализа специализированной текстовой информации.

4. Для решения задачи создания единых основ представления накопленных знаний и управления ими за счет интеграции и универсализации существующих способов систематизации таких знаний. Предлагается способ

преобразования знаний, приведенных к единому виду, при помощи моделей в стандартах серии *IDEF*, выбор компонентов которой осуществляется среди возможных решений на основе специально разработанных приемов, методик и типовых моделей организации системы и принятия решений.

5. Показано, что разработка новой модели управления и систематизации специализированной текстовой информации, и модернизация применяемых в ней методик, методов и алгоритмов системного анализа, искусственного интеллекта и компьютерной лингвистики позволяет повысить эффективность управления, систематизации и обработки специализированной текстовой информации больших объёмов с учетом особенностей русскоязычных текстовых документов.

6. Обоснован выбор методов для построения модели модернизированной системы систематизации и управления текстовой информацией. Для дальнейшего применения и модернизации в качестве базового выбран один из наиболее эффективных методов систематизации и управления информацией – *SVM*.

7. Разработана обобщенная модель управления и анализа информации, способствующая улучшению качества работы систем, решающих задачи этого класса, с целью оптимизации решения задачи управления и автоматической обработки специализированной информации. Благодаря применению разработанной модели появляется возможность оценить систему в состоянии равновесия и степень её чувствительности к различным факторам и внешним воздействиям, а также исследовать устойчивость поведения полученной модели в процессе принятия решений при обработке текстовой информации.

8. Для построения модернизированной модели управления и систематизации информации предложено использование оптимального способа анализа знаний, представляемого в виде стандарта *IDEF*. Формализация данных единым универсальным системным способом представления знаний позволяет создать соответствующие алгоритмы и подобрать инструментальные средства для обработки знаний с помощью единого формального аппарата.

9. Разработана обобщенная архитектура систематизации информации на основе предложенной модели, которая показывает общее представление ее построения и позволяет перейти к практической реализации ее прототипа и исследованию практической эффективности предложенных решений.

10. Для улучшения характеристик модели, осуществляющей систематизацию информации, предложены и обоснованы модели композиций методов систематизации, включающие в себя:

- удаление шумовых слов (позволяет уменьшить размер текстовых документов и повысит скорость работы);
- удаление окончаний, а также внесение изменений путем корректировки приоритетности слова в основном методе – *TF-IDF*, определяющий вес слова в тексте (позволяет более корректно распределить вес термов в документе, что в дальнейшем повысит качество работы классификатора);
- модернизацию метода *SVM* при помощи дополнения его алгоритмом дерево принятия решений с U – уровнями (позволяет повысить полноту и точность работы классификатора).

Использование предложенных композиций приводит к дополнительному влиянию на свойства предложенной модели управления и систематизации информации, и, как следствие, к повышению качества работы систем систематизации и управления текстовой информацией.

11. Экспериментальное исследование предложенных средств показало их позитивное влияние на усовершенствование предложенной общей модели систематизации и управления информацией на основе правил отбора неинформативных признаков, способов взвешивания термов и композиций методов систематизации.

12. Экспериментальное исследование моделей вычислительных композиций распределения веса слов в текстовом документе показало, что при разном распределении веса терма повышение качества систематизации

информации варьируется в пределах 10% в зависимости от используемой композиции.

13. Экспериментальное исследование модели классификации текстовой информации на основе предложенных композиций методов систематизации и внесения изменений в структуру алгоритма её построения показало, что полнота и точность работы модели повышается на 5,5% и 8,5 % соответственно.

14. Экспериментальное исследование предложенной общей модели автоматической систематизации и управления информацией, основанной на выполненном объединении достоинств синтагматических и парадигматических подходов, и всех предложенных модернизаций показало увеличение полноты и точности работы в среднем на 32,5% и 31,5% соответственно.

15. Экспериментальное исследование предложенной общей модернизированной модели управления и систематизации текстовых документов на основе всех разработанных средств показало повышение скорости и качества обработки текстовой информации в среднем более чем в 4 раза по сравнению с ручным способом (так на обработку одного тестового текстового документа вручную в среднем затрачивается 10 - 30 минут, в то время как автоматическая обработка на основе разработанных средств – в среднем затрачивает 3 - 7 минут).

16. Результаты исследований имеют широкий спектр применения для различных предметных областей. Предложенная практическая реализация усовершенствованной модели управления и систематизации информации позволяет формировать текстовые базы данных, содержащие классифицированную информацию, в автоматическом режиме. На основании результатов классификации появляется возможность автоматизировать работу специалистов-аналитиков, осуществляющих тематический анализ текстовой информации, и ведение аналитических задач в различных предметных областях, что может послужить функциональным дополнением и развитием информационных систем различных организаций.

СПИСОК ЛИТЕРАТУРЫ

1. Муна, Д. В. OPEN ACCESS: Системные риски для научного прогресса / Д. В. Муна, Е. В. Угриновича, В. А. Цветковой // [Электронный ресурс] XXI научно–практический семинар «Информационное обеспечение науки: новые технологии» 03–07 июня 2017. МЦНТИ–ICSTI 2017 – Режим доступа: http://www.benran.ru/SEM/Sb_17/present/pr_4.pdf. (по состоянию на 21.03.2019).
2. Фролов, А. В. Принцип конечной топологии распознавания топологических форм [Текст] / А. В. Фролов // Известия РАН. Теория и системы управления. – 2010. – №1. – С. 68–76.
3. Бринк, Х. Машинное обучение [Текст] / Х. Бринк, Дж. Ричардс, М. Феверолф; СПб.: Питер, 2017. – 336 с.
4. Nefedov, A. Support Vector Machines: A Simple Tutorial New York: / A. Nefedov // [Электронный ресурс] 2016. – P. 35. – Режим доступа: https://svmtutorial.online/download.php?file=SVM_tutorial.pdf (дата обращения: 21.03.2019).
5. Флах, П. Машинное обучение [Текст] / П. Флах; Наука и искусство построения алгоритмов, которые извлекают знания из данных. ДМК Пресс, 2015. – 402 с.
6. Российский семинар по Оценке Методов Информационного поиска [Электронный ресурс] Труды РОМИП 2010 Казань, 2010. – 214 с. – Режим доступа: http://romip.ru/romip2010/index.html (по состоянию на 21.03.2019).
7. Орельен, Ж. Прикладное машинное обучение с помощью Scikit-Learn и TensorFlow [Текст]/ Ж. Орельен; СПб.: Диалектика, 2019. – 683 с.
8. Ball, N. M. Data mining and machine learning in astronomy [Текст] / N. M. Ball, R. J. Brunner // International Journal of Modern Physics D. 2010. Vol. 19. – No. 7. – P. 1049–1106.
9. Farrar, C. R. Structural Health Monitoring: A Machine Learning Perspective [Текст] / C. R. Farrar, K. Worden // Wiley, 2013. – P. 643.

10. Dwyer, B. Systems Analysis and Synthesis: Bridging Computer Science and Information Technology [Текст] / Barry Dwyer; Morgan Kaufmann, 2016. – P 488.
11. Майер–Шенбергер, В. Большие данные. Революция, которая изменит то, как мы живем, работаем и мыслим [Текст] / В. Майер–Шенбергер, К. Кукьер; М. 2014. – 17 с.
12. Gantz, J. The Digital Universe in 2020: Big Data, Bigger Digital Shadows, and Biggest Growth in the Far East / J. Gantz and D. Reinsel; [Электронный ресурс] December 2012. – Режим доступа: http://www.emc.com/collateral/analyst–reports/idc–the–digitaluniverse–in–2020.pdf (дата обращения: 21.03.2019).
13. Dangeti, P. Statistics for Machine Learning [Текст] / P. Dangeti; Packt Publishing, 2017. – P. 450.
14. Sebastiani, F. Machine Learning in Automated Text Categorization [Текст] / F. Sebastiani // ACM Computing Surveys. –2012. –Vol. 34, – No. 1. – P. 47.
15. Ayodele, T. O. Types of Machine Learning Algorithms. New Advances in Machine Learning [Текст] / T. O. Ayodele // – INTECH Open Access Pub., 2010. P. 19–48.
16. Домингос, П. Верховный алгоритм. Как машинное обучение изменит наш мир [Текст] / Педро Домингос; М.: Манн, Иванов и Фербер, 2016. – 336 с.
17. Ibrahim, H. A. H. Taxonomy of Machine Learning Algorithms to classify realtime Interactive applications [Текст] / H. A. H. Ibrahim, S. M. Nor, A. Mohammed, A. B. Mohammed // International Journal of Computer Networks and Wireless Communications. 2012. Vol. 2. No. 1. P. 69–73.
18. Shoeb, A. H. Application of machine learning to epileptic seizure detection [Текст] / A. H. Shoeb , J. V. Guttag // Proceedings of the 27th International Conference on Machine Learning. 2010. P. 975–982.
19. Мюллер, А. Введение в машинное обучение с помощью Python [Текст] / А. Мюллер, С. Гидо; М.: O'Reilly Media, 2017. – 392 с.

20. Jolliffe, I. Principal Component Analysis / I. Jolliffe // [Электронный ресурс] Springer Series in Statistics, 2010. – P. 518 – Режим доступа: http://cda.psych.uiuc.edu/statistical_learning_course/Jolliffe%20I.%20Principal%20Component%20Analysis%20%282ed.,%20Springer,%202002%29%28518s%29_MVsa_.pdf (дата обращения: 21.03.2019).
21. Chandra, R. Parallel Programming in OpenMP / R. Chandra, R. Menon, L. Dagum, D. Kohr, D. Maydan, J. McDonald // [Электронный ресурс] Morgan Kaufmann, 2000. – P. 249 – Режим доступа: https://apps2.mdp.ac.id/perpustakaan/ebook/Karya%20Umum/Parallel_Programming_in_OpenMP.pdf (дата обращения: 21.03.2019).
22. Murphy, K. P. Machine Learning: A Probabilistic Perspective [Текст] / K. P. Murphy; Massachusetts Institute of Technology, 2012. – P. 1067.
23. Stewart, J. M. Python for Scientists [Текст] / J. M. Stewart; 2nd edition. — Cambridge: Cambridge University Press, 2017. – P 271.
24. Valacich, J. S. Modern Systems Analysis and Design [Текст] / J. S. Valacich, J. F. George // Boston: Pearson, 2016. P. – 545.
25. Саулин, Е. С. Зарождение и развитие искусственного интеллекта: характеристика исследовательских направлений. [Электронный ресурс] – / Е. С. Саулин // Режим доступа: http://journal.mrsu.ru/wp–content/uploads/2016/07/saulin.pdf (по состоянию на 21.03.2019).
26. Flach, P. Machine Learning: The Art and Science of Algorithms that Make Sense of Data [Текст] / P. Flach; Cambridge University Press, 2012. – P. 396.
27. Andrea, M. Machine learning methods for classifying human physical activity from on - body accelerometers [Текст] / M. Andrea, A. M. Sabatini // Sensors. 2013. Vol. 10. – No 2. – P.1154–1175.
28. Гифт, Н. Прагматичный ИИ. Машинное обучение и облачные технологии [Текст] / Ной Гифт; СПб.: Питер, 2019. – 304 с.
29. Tang, X. Understanding Online Consumer Review Opinions with Sentiment Analysis using Machine Learning [Текст] / X. Tang, C. Yang, Y. Wong, C. Wei

// Pacific Asia Journal of the Association for Information Systems. – 2010. –No 3(2). – P. 73–89.

30. Макаров, Р. И. Теория информационных процессов и систем [Текст] / Р. И. Макаров, Е. Р. Хорошева; Учеб. пособие. – Владимир: Изд–во ВлГУ, 2018. – 175 с.
31. Ипатова, Э. Р. Методологии и технологии системного проектирования информационных систем [Текст] / Э. Р. Ипатова; Учебник: моногр. – М.: Флинта, 2016. – 300 с.
32. Cooling, J. Modelling software with pictures: Practical UML diagramming for real-time systems [Текст] / J. Cooling; Lindentree Associates, 2015. – P. 365.
33. Черемных, С. В. Моделирование и анализ систем. IDEF–технологии [Текст] / С. В. Черемных, И. О. Семенов, В. С. Ручкин; Учебник–практикум. – М.: Финансы и статистика, 2006. – 188 с.
34. Голицына, О. Л. Информационные системы [Текст] / О. Л. Голицына, Н. В. Максимов, И. И. Попов; Учебное пособие -2-е изд. – М.: Форум: НИЦ ИНФРА-М, 2014. – 448 с.
35. Иванов, Д. Ю. Унифицированный язык моделирования UML [Текст] / Д. Ю. Иванов, Ф. А. Новиков; Учебное пособие. – СПб.: Изд-во Политехн. ун-та, 2010. – 249 с.
36. Горлушкина, Н. Н. Системный анализ и моделирование информационных процессов и систем [Текст] / Н. Н. Горлушкина; учебное пособие. - СПб: Университет ИТМО, 2016. –120 с.
37. Бахтизин, В. В. Технологии разработки программного обеспечения [Текст] / В. В. Бахтизин, Л. А. Глухова; Учебное пособие. – Минск: БГУИР, 2010. – 267 с.
38. Душин, В. К. Теоретические основы информационных процессов и систем [Текст] / В. К. Душин; Учебник – 5-е изд. – М.: Издательско-торговая корпорация "Дашков и К-", 2014. – 348 с.

39. Цуканова, О. А. Методология и инструментарий моделирования бизнес–процессов [Текст] / О. А. Цуканова; учебное пособие – СПб.: Университет ИТМО, 2015. – 100 с.
40. Волкова, В. Н. Теория систем и системный анализ [Текст] / В. Н. Волкова, А. А. Денисов; – М.: Юрайт, 2014. – 616 с.
41. Ипатова, Э. Р. Методологии и технологии системного проектирования информационных систем [Текст] / Э. Р. Ипатова; Учебник: моногр. – М.: Флинта, 2016. – 300 с.
42. Дворников, А. IDEF0 как инструмент моделирования процессов [Текст] / А. Дворников; Авант Партнер, – 2005. – № 22. – 79 с.
43. Батин, Н. В. Основы автоматизированного управления [Текст] / Н. В. Батин; Учебное пособи. – Минск: Белорусский государственный университет информатики и радиоэлектроники, 2006. – 68 с.
44. Горбаченко, В. И. Проектирование информационных систем с CA ERwin Modeling Suite 7.3 [Текст] / В. И. Горбаченко, Г. Ф. Убиенных, Г. В. Бобрышева; Учебное пособие. – Пенза: Изд-во ПГУ, 2012. – 154 с.
45. Абраменко, Г. В. Практические рекомендации по применению системного анализа к проектированию сложных систем [Текст] / Г. В. Абраменко, К. В. Власов, М. А. Краснощеков; – Москва: Оргсервис-2000, 2015. – 300 с.
46. Андрейчиков, А. В. Системный анализ и синтез стратегических решений в инноватике. Математические, эвристические и интеллектуальные методы системного анализа и синтеза инноваций [Текст] / А. В. Андрейчиков, О. Н. Андрейчикова; Изд. 2-е.– М.: URSS: [Книжный дом ЛИБРОКОМ"], 2013. – 304 с.
47. Рассел, С. Искусственный интеллект. Современный подход [Текст] / С. Рассел, П. Норвиг; 2е издание. – Вильямс, 2015. – 1407 с.
48. Козлов, В. Н. Системный анализ, оптимизация и принятие решений [Текст] / В.Н. Козлов; – М.: Проспект, 2016. – 176 с.

49. Новосельцев, В. И. Теоретические основы системного анализа [Текст] / В. И. Новосельцев, Б. В. Тарасов; Изд. 2-е, исправленное и переработанное; под ред. В.И. Новосельцева. – М.: Майор, 2013. – 536 с.
50. Дошина, А. Д. Экспертная система. Классификация. Обзор существующих экспертных систем / А. Д. Дошина // [Электронный ресурс] Молодой ученый. – 2016. – №21. – С. 756-758. – Режим доступа: https://moluch.ru/archive/125/34485/ (дата обращения: 21.03.2019).
51. Вдовин, В. М. Теория систем и системный анализ [Текст] / В. М. Вдовин, Л. Е. Суркова; Учебник для бакалавров - М.: Дашков и К, 2016. – 644 с.
52. Андрейчиков, А. В. Системный анализ и синтез стратегических решений в инноватике: Концептуальное проектирование инновационных систем [Текст] / А. В. Андрейчиков, О. Н. Андрейчикова; – М.: Ленанд, 2014. – 432 с.
53. Санников, А. А. Системный анализ при принятии решений [Текст] / А. А. Санников, Н. В. Куцубина; Учебное пособие. – Екатеринбург: Урал. гос. лесотехн. ун-т, 2015. – 137 с.
54. Бурлаева, Е. И. Анализ методов преобразования текстов в форму объектов векторного пространства [Текст] / Е. И. Бурлаева, В. Н. Павлыш // «Программная инженерия», Т. 10, – №1. – Москва, 2019. – С 30–37. 132
55. Большакова, Е. И. Автоматическая обработка текстов на естественном языке и компьютерная лингвистика [Текст] / Е. И. Большакова, Э. С. Клышинский, Д. В. Ландэ, А. А. Носков, О. В. Пескова, Е. В. Ягунова; Учеб. пособие: М.: МИЭМ, 2011. – 272 с.
56. Бурлаева, Е. И. Обзор методов классификации текстовых документов на основе подхода машинного обучения [Текст] / Е. И Бурлаева // «Программная инженерия», Т. 8, – №7. – Москва, 2017. – С 328–336. 126
57. Николаев, И. С. Прикладная и компьютерная лингвистика [Текст] / И. С. Николаев, О. В. Митренина, Т. М. Ландо; – М.: ЛЕНАНД, 2016. – 316 с.
58. Захаров, В. П. Корпусная лингвистика / В. П. Захаров, С. Ю. Богданова // [Электронный ресурс] Учебник Иркутск ИГЛУ 2011. – 161 с. – Режим

доступа: http://lazareva.hol.es/Zaharov,Bogdanova.pdf (дата обращения: 21.03.2019).

59. Чапайкина, Н. Е. Семантический анализ текстов. Основные положения / Н. Е. Чапайкина // [Электронный ресурс] Молодой ученый. – 2012. – №5. – С. 112–115. – Режим доступа: https://moluch.ru/archive/40/4857/ (дата обращения: 21.03.2019).
60. Сова, Л. З. Аналитическая лингвистика и типология [Текст] / Л. З. Сова; СПб.: Изд-во Политехн. ун-та, 2007. – 378 с.
61. Ляшевская, О. Н. Корпусные инструменты в грамматических исследованиях русского языка / О. Н. Ляшевская // [Электронный ресурс] – М.: Издательский дом ЯСК, Рукописные памятник Древней Руси, 2016. – 520 с. – Режим доступа: https://publications.hse.ru/mirror/pubs/share/folder/omum0edrwg/direct/182339609 (дата обращения: 21.03.2019).
62. Боярский, К. К. Введение в компьютерную лингвистику [Текст] / К. К. Боярский; Учебное пособие. – СПб: НИУ ИТМО, 2013. – 72 с.
63. Копотев, М. В. Введение в корпусную лингвистику [Текст] / М. В. Копотев; Учебное пособие для студентов филологических и лингвистических специальностей университетов. — Прага: Animedia Company, 2014. — 230 с.
64. Склярова, Н. Г. Введение в прикладную лингвистику. Информационные технологии в лингвистике [Текст] / Н. Г. Склярова; Учебное пособие. – Пятигорск: Изд–во ПГУ, 2016. – 86 с.
65. Болховитянов, А. В. Алгоритмы морфологического анализа компьютерной лингвистики [Текст] / А. В. Болховитянов, А. М. Чеповский; учеб. пособие. М.: МГУП имени Ивана Федорова, 2013. – 198 с.
66. Шипицына, Л. Ю. Информационные технологии в лингвистике [Текст] /Л. Ю. Щипицина; учеб. пособие: — М.: ФЛИНТА: Наука, 2013. — 128с.
67. Царьков, С. В. Автоматическое выделение ключевых фраз для построения словаря терминов в тематических моделях коллекций текстовых

документов [Текст] / С. В. Царьков // Естественные и технические науки. – 2012. – №6. – С. 456–464.

68. Вьюгин, В. В. Математические основы машинного обучения и прогнозирования [Текст] / В. В. Вьюгин; — М.: МЦНМО, 2014. – 304 с.
69. Рашид, Т. Создаем нейронную сеть [Текст] / Т. Рашид; СПб.: Альфа-книга, 2017. – 274 с.
70. Schmidhuber, J. Deep Learning in Neural Networks: An Overview [Текст] / J. Schmidhuber // Neural Networks, Vol 61, Jan 2015. – P. 85–117.
71. Сосина, Е. П. Введение в прикладную лингвистику [Текст] / Е. П. Соснина; учебное пособие – 2-е изд., испр. и доп. – Ульяновск: УлГТУ, 2012. – 110 с.
72. Сироткин, А. В. Тематическая модель рейтингования интернет–сайтов по критерию социальной значимости / А. В. Сироткин, С. А. Шарыпов // [Электронный ресурс] Электронный научный журнал «Инженерный вестник Дона» том 43 – №4 2016. – Режим доступа: https://cyberleninka.ru/article/n/tematicheskaya–model–reytingovaniya–internet–saytov–po–kriteriyu–sotsialnoy–znachimosti (дата обращения: 21.03.2019).
73. Недильченко, О. С. Этапы и методы автоматического извлечения ключевых слов / О. С. Недильченко // [Электронный ресурс] Молодой ученый. – 2017. – №22. – С. 60-62. – Режим доступа: https://moluch.ru/archive/156/44044/ (дата обращения: 21.03.2019).
74. Rajaraman, A. Data Mining / A. Rajaraman, J. D. Ullman // [Электронный ресурс] Mining of Massive Datasets. 2011. – P. 1–17. – Режим доступа: http://i.stanford.edu/~ullman/mmds/ch1.pdf (дата обращения: 21.03.2019).
75. Коршунов, А. Тематическое моделирование текстов на естественном языке [Текст] / А. Коршунов, А. Гомзин; Труды Института Системного Программирования Ран, Выпуск том 23. – 2012. – С. 216–243.
76. Губин, М. В. Модели и методы представления текстового документа в системах информационного поиска [Текст] / М. В. Губин // Научно-техническая информация. Сер. 1. – 2014. – № 12. – С. 12-24.

77. Михайлов, Д. В. Выделение знаний и языковых форм их выражения на множестве тематических текстов: подход на основе меры TF-IDF [Текст] / Д. В. Михайлов, А. П. Козлов, Г. М. Емельянов // Компьютерная оптика. – 2015. Т. 39. – №3. – С. 429–438.
78. Mowafy, M. An Efficient Classification Model for Unstructured Text Document / M. Mowafy, A. Rezk, H. M. El-bakry // [Электронный ресурс] Am J Compt Sci Inform Technol Vol.6 – No.1: 16 – 2018. – P 10 – Режим доступа: http://www.imedpub.com/articles/an-efficient-classification-model-for-unstructured-text-document.pdf (дата обращения: 21.03.2019).
79. Xin, R. Word2vec Parameter Learning Explained / Rong. Xin // [Электронный ресурс] Jun 2016. – Режим доступа: https://arxiv.org/pdf/1411.2738v4.pdf (дата обращения: 21.03.2019).
80. Загорулько, Ю. А. Классификация деловых писем в системе документооборота [Текст] / Ю. А. Загорулько, И. С. Кононенко, Ю. В. Костов, Е. В. Сидорова // Международная конференция ИСТ'2003 «Информационные системы и технологии» – Новосибирск, 2013. – С. 87–110.
81. Будников, Е. А. Обзор некоторых статистических моделей естественных языков. Машинное обучение и анализ данных / Е. А Будников // [Электронный ресурс] 2011. Т. 1. – № 2 – Режим доступа: http://jmlda.org/papers/doc/2011/no2/Budnikov11Statistical.pdf (дата обращения: 21.03.2019).
82. Sidorov, G. Syntactic dependency based N-grams in rule based automatic English as second language grammar correction [Текст] / G. Sidorov // International Journal of Computational Linguistics and Applications. – 2013. – Vol. 4(2). – P. 169–188.
83. Михайлов, Д. В. Выделение знаний и языковых форм их выражения на множестве тематических текстов анализом связей слов в составе n-грамм [Текст] / Д. В. Михайлов, А. П. Козлов, Г. М. Емельянов // Компьютерная оптика. – 2017. – Т. 41, № 3. – С. 461–471.

84. Попков, М. И. Автоматическая система классификации текстов для базы знаний предприятия / М. И. Попков // [Электронный ресурс] International Journal of Open Information Technologies vol. 2, – no. 7. 2014. – Режим доступа: http://injoit.org/index.php/j1/article/viewFile/118/91 (дата обращения: 21.03.2019).
85. Игнатьев, Н. А. Интеллектуальный анализ данных на базе непараметрических методов классификации и разделения выборок объектов поверхностями [Текст] / Н. А Игнатьев; Монография. – Ташкент: Национальный университет Узбекистана им. Мирзо Улугбека, 2010. – 140 с.
86. Бородкин, А. А. Разработка и исследование методов взвешивания ближайших соседей (на примере классификации библиографических текстовых документов) [Текст] / А. А. Бородкин, В. О. Толчеев // Заводская лаборатория. Диагностика материалов. 2013. Т.79. – №7. С.70–74.
87. Воронцов, К. В. Математические методы обучения по прецедентам (теория обучения машин) / К. В. Воронцов // [Электронный ресурс] Москва, 2011. – Режим доступа: http://www.machinelearning.ru/wiki/images/6/6d/Voron–ML–1.pdf (дата обращения: 21.03.2019).
88. Воронцов, К. В. Многокритериальные и многомодальные вероятностные тематические модели коллекций текстовых документов [Текст] / К. В. Воронцов, А. А. Потапенко, А. И. Фрей, М. А. Апишев, Н. В. Дойков, А. В. Шапулин, Н. А. Чиркова; 10-я Междунар. конф. ИОИ-2014: Тезисы докладов. – 2014. – 198 с.
89. Davidson-Pilon, C. Bayesian Methods for Hackers: Probabilistic Programming and Bayesian Inference [Текст] / Cameron, Davidson-Pilon; Addison-Wesley Data & Analytics, 2015. – P. 256.
90. Kruschke, J. K. Doing Bayesian Data Analysis, Second Edition: A Tutorial with R, JAGS, and Stan [Текст] / John K. Kruschke; Academic Press / Elsevier, 2015. – P. 748.

91. LeCun, Y. Deep learnin [Текст] / Yann LeCun, Yoshua Bengio, Geoffrey Hinton // Nature 521, (28 May 2015). – P. 436–444.
92. Епрев, А. С. Автоматическая классификация текстовых документов [Текст] / А. С. Епрев // Математические структуры и моделирование. – 2010. – № 21. – С. 65–81.
93. Ma, Y. (Eds.) Support Vector Machines Applications [Текст] / Y. Ma, G. Guo; Springer Cham Heidelberg New York Dordrecht London, 2014, VII, – P. 302.
94. Moraes, R. Document level sentiment classification: An empirical comparison between SVM and ANN [Текст] / R. Moraes, J. F. Valiati, and W. P. Gavião Neto // Expert Systems with Applications, 2013, – no. 40, – P. 621–633.
95. Liwei, Wei. Text Classification Using Support Vector Machine with Mixture of Kernel / Wei Liwei, Wei Bo, Wang Bin // [Электронный ресурс] A Journal of Software Engineering and Applications, 2012, – № 5, – P. 55–58. – Режим доступа: http://file.scirp.org/pdf/JSEA_2013011816511264.pdf (дата обращения: 21.03.2019).
96. Воронина, В. В. Теория и практика машинного обучения [Текст] / В. В. Воронина, А. В. Михеев, Н. Г. Ярушкина, К. В. Святов // Учебное пособие. Теория + Практикум (листинги) – Ульяновск: УлГТУ, 2017. – 290 с.
97. Andri, R. Design of Fuzzy Rule–based Classifiers through Granulation and Consolidation / Riid Andri, Preden Jürgo–Sören // [Электронный ресурс] DE Grouter open Jaiscr, 2017, Vol.7, – No 2, – P. 137–147. – Режим доступа: https://doi.org/10.1515/jaiscr–2017–0010 (дата обращения: 21.03.2019).
98. Badr, H. A comparative study of decision tree ID3 and C4.5 / Hssina Badr, Merbouha Abdelkarim, Ezzikouri Hanane, Erritali Mohammed. // [Электронный ресурс] – Режим доступа: https://saiconference.com/Downloads/SpecialIssueNo10/Paper_3-A_comparative_study_of_decision_tree_ID3_and_C4.5.pdf (дата обращения: 21.03.2019).
99. Marco, A. Classification ofheart disease using multiple classifiers / Alfonse Marco // [Электронный ресурс] Review of Computer Engineering Studies

Vol.5, – No.3, September, 2018, – P. 45-49 – Режим доступа: http://iieta.org/sites/default/files/Journals/RCES/05.03_01.pdf (дата обращения: 21.03.2019).

100. Behera, H. S. Computational Intelligence in Data Mining. Proceedings of the International Conference on ICCIDM 2018 [Текст] / H. S. Behera, J. Nayak, B. Naik, D. Pelusi, (Eds.); Springer, 2018. – P. 895.
101. Чистяков, С. П. Случайные Леса: Обзор / С. П. Чистяков // [Электронный ресурс] Труды Карельского научного центра РАН. 2013. – № 1. С. 117 – 136 – Режим доступа: http://resources.krc.karelia.ru/transactions/doc/trudy2013/trudy_2013_1_117-136.pdf (дата обращения: 21.03.2019).
102. Кашницкий, Ю. С. Ансамблевый метод машинного обучения, основанный на рекомендации классификаторов / Ю. С. Кашницкий, Д. И. Игнатов // [Электронный ресурс] – Режим доступа: http://intsysjournal.ru/articles/is1904/02_kashnickiy.pdf (дата обращения: 21.03.2019).
103. Anil, K. J. Statistical pattern recognition: A review / K. Jain Anil, P. Robert, W. Duin, and Jianchang Mao // [Электронный ресурс] IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, – no. 1, 2000. – P. 4 –37. – Режим доступа: http://www.cs.bilkent.edu.tr/~saksoy/courses/cs551-Spring2009/papers/jain00_pr_survey.pdf (дата обращения: 21.03.2019).
104. Sewell, M. Ensemble Learning / Martin Sewell // [Электронный ресурс] Research Note 20 January 2011 P 12 – Режим доступа: http://www.cs.ucl.ac.uk/fileadmin/UCL-CS/research/Research_Notes/RN_11_02.pdf (дата обращения: 21.03.2019).
105. Spirin, N. V. Learning to Rank with Nonlinear Monotonic Ensemble [Текст] / N. V. Spirin, K. V. Vorontsov // В кн.: 10th International Workshop on Multiple Classidier Systems. Naples, Italy, June 15–17, 2011. – Lecture Notes in Computer Science. Springer-Verlag, 2011. – P. 16-25.

106. Альсова, О. К. Неоднородный ансамблевый алгоритм классификации разнотипных данных [Текст] / О. К. Альсова, И. М. Стубарев // Известия Самарского научного центра Российской академии наук, Т. 19. – 2017. – № 6. – С. 118–123.
107. Jonsson, L. Automated bug assignment: Ensemble-based machine learning in large-scale industrial contexts [Текст] / L. Jonsson, M. Borg, D. Broman et al.; Empirical Software Engineering – 2014. – P. 53.
108. Izenman, A. J. Modern Multivariate Statistical Techniques, Springer Texts in Statistics [Текст] / A. J. Izenman; Springer Science+Business Media New York, 2013. – P. 757.
109. Batygin, R. I. Software system for different types of data classification based on the ensemble algorithms [Текст] / R. I. Batygin, O. K. Alsova // Actual problems of electronic instrument engineering (APEIE–2016) : proceedings. Novosibirsk, 2016. V. 1. Part 2. – P. 506–509.
110. Михайлов, Д. В. Выделение знаний, языковых форм их выражения и оценка эффективности формирования множества тематических текстов [Текст] / Д. В. Михайлов, А. П. Козлов, Г. М. Емельянов // Компьютерная оптика. – 2016. – Т. 40. – № 4. – С. 572–582.
111. Мухамедиев, Р. И. Таксономия методов машинного обучения и оценка качества классификации и обучаемости / Р. И. Мухамедиев, Е. Л. Мухамедиева, Я. И. Кучин // [Электронный ресурс] Cloud of Science. 2015. Т. 2. – № 3. С. – 359–378 – Режим доступа: https://cyberleninka.ru/article/v/taksonomiya-metodov-mashinnogo-obucheniya-i-otsenka-kachestva-klassifikatsii-i-obuchaemosti (дата обращения: 21.03.2019).
112. Ashurov, M. F. Text classification stream-based r-measure approach using frequency of substring repetition / M. F. Ashurov, V. V. Poddubny // [Электронный ресурс] Управление, вычислительная техника и информатика – № 4 (33) 2015. – С 4–12 – Режим доступа: https://cyberleninka.ru/article/v/text-classification-stream-based-r-measure-

approach-using-frequency-of-substring-repetition (дата обращения: 21.03.2019).

113. Агеев, М. С. Подготовка Web–версий традиционных изданий [Текст] / М. С. Агеев, С. В. Журавлев, В. Г. Ламбурт // Открытые Системы, – 2000. – №12. – 5 с.
114. Павлов, Ю. Н. Сравнение методов оценки тональности текста / Ю. Н. Павлов, К. А. Майструк // [Электронный ресурс] Молодой ученый. – 2016. – №12. – С. 59-64. – Режим доступа: https://moluch.ru/archive/116/31521/ (дата обращения: 21.03.2019).
115. Агеев, М. С. Метод машинного обучения, основанный на моделировании логики рубрикатора / М. С. Агеев, Б. В. Добров, Н. В. Макаров–Землянский // [Электронный ресурс] RCDL'2003 Электронные библиотеки: перспективные методы и технологии, электронные коллекции: Пятая всероссийская науч. конф. – Санкт–Петербург, 2013. – Режим доступа: http://rcdl.ru/doc/2003/B2.pdf (дата обращения: 21.03.2019).
116. Deng, H. «Bias of importance measures for multi-valued attributes and solutions» [Текст] / H. Deng, G. Runger, E. Tuv // Proceedings of the 21st International Conference on Artificial Neural Networks (ICANN). 2011. – P. 293–300.
117. Лукашевич, Н. В. Тезаурусы в задачах информационного поиска [Текст] / Наталья Валентиновна Лукашевич; Москва Тезаурусы в задачах информационного поиска – М., 2010. – 396 с.
118. Агеев, М. С. Автоматическая рубрикация текстов: методы и проблемы [Текст] / М. С. Агеев, Б. В. Добров, Н. В. Лукашевич // Ученые записки Казанского Государственного Университета. Серия Физико–математические науки. – 2008. –Т. 150, Кн. 4. – С. 25–40.
119. Ageev, M. On–line Thematic and Metadata Analysis of Document Collection [Текст] / M. Ageev, B. Dobrov, N. Makarov–Zemlyanskii // New Trends in Intelligent Information Processing and Web Mining’2004: Proceedings of the

International Conference / Springer,Advanced in Soft Computing – Zakopane, Poland, May 2004. – P. 279–286.

120. Keji, W. The Comparison of Machine Learning Algorithms on Online Classification of Network Flows [Текст] / Wei Keji, Cao Shaolong, Yu Jian; International Journal of Wireless and Microwave Technologies(IJWMT) @ijwmt 2 Vol.2, 2012.
121. Rose, T. The Reuters Corpus Volume 1 – from Yesterday News to tomorrow's Language [Текст] / T. Rose, M. Stevenson, M. Whitehead // In Proceedings of the Third International Conference on Language Resources and Evaluation, Las Palmas de Gran Canaria, May 2002. – P. 29–31.
122. Похорукова, М. Ю. Метод автоматической классификации документов в задаче профессионального самоопределения / М. Ю. Похорукова, В. М. Самохина // [Электронный ресурс] Молодой ученый. – 2016. – №11. – С. 40-43. – Режим доступа: https://moluch.ru/archive/115/30942/ (дата обращения: 21.03.2019).
123. Драль, А. А. Классификация коротких текстовых документов [Текст] / А. А. Драль, Э. Мбайкоджи // [Электронный ресурс] Информационно телекоммуникационные технологии и математическое моделирование высокотехнологичных систем: Тезисы докладов Всероссийской конференции с международным участием – М.: РУДН. – 2012. – С. 121–123.
124. Мбайкоджи, Э. Метод автоматической классификации коротких текстовых сообщений / Э. Мбайкоджи, А. А. Драль, И. В. Соченков // [Электронный ресурс] Информационные технологии и вычислительные системы 3/2012 – Режим доступа: http://www.isa.ru/jitcs/images/documents/2012-03/93_102.pdf обращения: 21.03.2019).
125. Бурлаева, Е. И. Проект построения алгоритма классификации текстовых документов [Текст] / Е. И. Бурлаева, В. Н. Павлыш // Проблемы искусственного интеллекта, – №4 (7). – Донецк, 2017. – С 24–32.

126. Батура, Т. В. Математическая лингвистика и автоматическая обработка текстов на естественном языке [Текст] / Т. В. Батура; Учебное пособие. – Новосибирск: РИЦ НГУ, 2016. – 166 с.
127. Орлов, С. А. Технологии разработки программного обеспечения [Текст] / С. А. Орлов, Б. Я. Цилькер; 4-е изд. – СПб.: Питер, 2012. – 608 с.
128. Grashhenko, L. A. O model'nom stop–slovare [Текст] / L. A. Grashhenko // Izvestija Akademii nauk Respubliki Tadzhikistan. Otdelenie fiziko–matematicheskih, himicheskih, geologicheskih i tehnicheskih nauk –2013. – № 1(150). – P. 40–46.
129. Gubin, M. V. Vlijanie morfologicheskogo analiza na kachestvo informacionnogo poiska [Текст] / M. V. Gubin, A. B. Morozov; Konsorcium «Kodeks». 2006. – P. 16.
130. Norkin, V. On stochastic optimization and statistical learning in reproducing kernel Hilbert spaces by Support Vector Machines (SVM) [Текст] / V. Norkin, M. Keyzer // Informatica. 2009. Vol. 20, – No. 2. – P. 273–292.
131. Болховитянов, А. В. Алгоритмы морфологического анализа компьютерной лингвистики [Текст] / А. В. Болховитянов, А. М. Чеповский; учеб. пособие. М.: МГУП имени Ивана Федорова, 2013. – 198 с.
132. Некрестьянов, И. С. Оценка систем текстового поиска [Текст] / И. С. Некрестьянов, И. Е. Кураленок // Программирование, Москва, Россия – 2012. – 28 (4). – С. 226–246.
133. Павлыш, В. Н. Комбинированный подход к решению задач классификации текстовых массивов [Текст] / В. Н. Павлыш, Е. И Бурлаева // Материалы V Международной научно–технической конференции «Современные информационные технологии в образовании и научных исследованиях» (СИТОНИ–2017). – Донецк: ДонНТУ, 2017. – 442с.
134. Suthaharan, S. Machine Learning Models and Algorithms for Big Data Classification. Thinking with Examples for Effective Learning [Текст] / S. Suthaharan; Springer, 2016. – P. 364.

135. Павлыш, В. Н. Задача классификации информации при формировании баз данных в компьютерных обучающих системах [Текст] / В. Н. Павлыш, С. А. Зори, Е. И. Бурлаева // Проблемы искусственного интеллекта, – №4 (11). – Донецк, 2018. – С 71–81.
136. Павлыш, В. Н. Системный анализ и векторизация текстовой информации [Текст] / В. Н. Павлыш, Е. И. Бурлаева, С. А. Зори // Машиностроение и техно сфера XXI века; Сборник трудов XXV международной научно–технической конференции в г. Севастополе 10–16 сентября 2018 г. В 2–х томах. – Донецк: ДонНТУ, 2018. Т. 2. – С 32–37.
137. Бурлаева, Е. И. Сравнение некоторых методов машинного обучения для анализа текстовых документов [Текст] / Е. И. Бурлаева, С. А. Зори // Проблемы искусственного интеллекта, – №1 (12). – Донецк, 2019. – С 42–51.
138. Макарова, И. О. Компьютерная графика в книжной иллюстрации / Макарова И. О // [Электронный ресурс] Вестник Адыгейского государственного университета. Серия 2: Филология и искусствоведение 2011. С. 5. – Режим доступа: https://cyberleninka.ru/article/n/kompyuternaya-grafika-vknizhnoy-illyustratsii.pdf (дата обращения: 21.03.2019).
139. Бурлаева, Е. И. Сопоставление методов автоматической обработки текста [Текст] / Е. И. Бурлаева, Т. В. Ермоленко // Информатика, управляющие системы, математическое и компьютерное моделирование в рамках III форума «Информационные перспективы Донбасса» (ИУСМКМ – 2017): VIII Международная научно–техническая конференция, 25 мая 2017, г. Донецк: / Дон.нац. техн. ун–т; редкол. Ю. К. Орлов и др. – Донецк: ДонНТУ, 2017. – 802 с.
140. Бурлаева, Е. И. Анализ работы классификаторов на русскоязычном массиве документов [Текст] / Е. И. Бурлаева // Донбасс будущего глазами молодых ученых, г. Донецк, 20 ноября 2018 г. – Донецк: ДонНТУ, 2018. – 264 с.

Приложение А

Копии документов о внедрении результатов исследований

ДОНЕЦКАЯ НАРОДНАЯ РЕСПУБЛИКА
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
"ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ"
283001, г. Донецк, ул. Артема, 58 тел.: (062) 337-17-33, 335-75-62, факс: (062) 304-12-78
эл. почта: donntu.info@mail.ru

14.05.2019 №52.1-05/19
На № _____

СПРАВКА
о внедрении результатов исследований диссертационной работы
Бурлаевой Екатерины Игоревны «Совершенствование методов системного анализа в задачах управления и систематизации специализированной информации», представленной на соискание ученой степени кандидата технических наук по специальности 05.13.01 – «Системный анализ, управление и обработка информации» (по отраслям), (технические науки).

В ГОУ ВПО «Донецкий национальный технический университет» приняты к внедрению в учебный процесс и используются при чтении лекций и проведении практических (лабораторных) занятий для подготовки бакалавров на кафедрах «Искусственный интеллект и системный анализ» (ИИСА) и «Прикладная математика» (ПМ) следующие разработки, полученные в диссертационной работе Бурлаевой Е.И.:

- предложенный способ преобразования знаний, приведенных к единому виду при помощи моделей в стандартах серии IDEF, используется для решения задачи создания единых основ представления накопленных знаний и управления ими за счет интеграции и универсализации существующих способов систематизации (направление подготовки 09.03.02 «Информационные системы и технологии», 09.03.03 «Прикладная информатика», дисциплины: «Организация баз данных и знаний», «Стандартизация и сертификация в сфере информационных технологий», н.п. 01.03.04 «Прикладная математика», дисц. «Базы данных»);
- обоснованный выбор методов для построения модели модернизированной системы систематизации и управления текстовой информацией используется для дальнейшего применения и модернизации информационных систем, в качестве базового рекомендован один из наиболее эффективных методов систематизации и управления информацией – *SVM* (н.п. 09.03.03 «Прикладная информатика», дисц. «Распределённые информационно-аналитические системы»);
- разработана обобщенная архитектура систематизации информации на основе предложенной модели, которая показывает общее представление ее построения и позволяет перейти к практической реализации рассматриваемых решений (н.п. 09.03.02 «Информационные системы и технологии», дисц. «Корпоративные информационные системы»);
- реализация усовершенствованной модели управления и систематизации информации в виде программного пакета, что позволяет формировать текстовые базы данных, содержащие классифицированную информацию, в автоматическом режиме (09.03.02 «Информационные системы и технологии», дисц. «Стандартизация и сертификация в сфере информационных технологий»).

Первый проректор
ГОУ ВПО «Донецкий национальный
технический университет» А.А. Каракозов

Нач. учебного отдела Б.В. Гавриленко

Зав. каф. ИИСА А.С. Миненко

Зав. каф. ПМ В.Н. Павлыш

М.П.

ДОНЕЦКАЯ НАРОДНАЯ РЕСПУБЛИКА
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РЕСПУБЛИКАНСКИЙ АКАДЕМИЧЕСКИЙ НАУЧНО-ИССЛЕДОВАТЕЛЬСКИЙ И ПРОЕКТНО-КОНСТРУКТОРСКИЙ ИНСТИТУТ ГОРНОЙ ГЕОЛОГИИ, ГЕОМЕХАНИКИ, ГЕОФИЗИКИ И МАРКШЕЙДЕРСКОГО ДЕЛА (РАНИМИ)

Донецкая Народная Республика
83004, город Донецк, Киевский район
улица Челюскинцев, 291
Тел.: +38 (062) 300 27 91; Тел/факс +38 (062) 300 27 92
E-mail: ranimi@ranimi.org

20.05.2019 № 01/140
на № _____ от _____

Соискателю учёной степени кандидата технических наук по специальности 05.13.01 – Системный анализ, управление и обработка информации (по отраслям)

Бурлаевой Екатерине Игоревне

С П Р А В К А

о внедрении результатов исследований диссертационной работы Бурлаевой Екатерины Игоревны «Совершенствование методов системного анализа в задачах управления и систематизации специализированной информации», представленной на соискание ученой степени кандидата технических наук по специальности 05.13.01 – Системный анализ, управление и обработка информации (по отраслям), (технические науки).

В отделе сдвижения земной поверхности и охраны подрабатываемых объектов (СЗПО) Республиканского академического научно-исследовательского и проектно-конструкторского института горной геологии, геомеханики, геофизики и маркшейдерского дела (РАНИМИ) с целью дальнейшего развития работ по совершенствованию компьютерной технологии прогноза сдвижений и деформаций земной поверхности от влияния подземных горных работ приняты к использованию следующие разработки и рекомендации, полученные в диссертационной работе Бурлаевой Е.И.:

- модель управления и систематизации специализированной текстовой информации для решения задачи создания единых основ представления накопленных знаний и управления ими, что способствует оптимизации процесса систематизации и обработки специализированной текстовой информации больших объёмов с учетом особенностей текстовых документов;

- обобщенная архитектура систематизации информации на основе предложенной модели, которая показывает общее представление ее построения и позволяет перейти к практической реализации предложенных решений;

- реализация усовершенствованной модели управления и систематизации информации в виде программного пакета.

Использование результатов указанных разработок позволяет формировать текстовые базы данных, содержащие классифицированную информацию, в автоматическом режиме, а также обеспечивает на основании результатов классификации возможность автоматизации работы специалистов, осуществляющих тематический анализ текстовой информации и решение аналитических задач в различных предметных областях, в том числе в области прогноза сдвижений и деформаций земной поверхности от влияния подземных горных работ. Применение результатов разработок Бурлаевой Е.И. может послужить функциональным дополнением и развитием системы поддержки принятия решений при подготовке проектов подработки зданий и сооружений, которая создаётся в отделе СЗПО.

Зам. директора по научной работе,
д-р техн. наук  Дрибан В.А.

Исп. Грищенков Н.Н.
тел. +38 (071) 334-90-30

Приложение Б

Список стоп–слов

а, аа, а–а, ааа, а–а–а, а–а–а–а, абы, авось, ага, аж, аз, ай, ай–ай–ай, айда, ай–яй–яй, аки, але, али, алле, алло, аль, а–ля, аминь, ан, апчхи, атас, ау, аф, ах, ахти, аще, б, ба, бабах, ба–бах, баста, бах, бац, без, безо, бен, бис, бишь, благо, благодаря, близ, блин, бляха–муха, бо, боже, более, больше, бом, браво, брр, бррр, брысь, бу–бу–бу, буде, будто, буль–буль, бум, бы, было, быть, в, вай, ван, вау, ваш, вблизи, ввиду, вглубь, вдоль, ведь, ведь, везде, весь, взамен, виват, вишь, включая, вкруг, вместо, вне, внизу, внутри, внутрь, во, во–во, возле, вокруг, вон, вон, вона, вообще–то, во–он, во–от, вопреки, восемнадцатый, восемнадцать, восемь, восемьдесят, восемьсот, вослед, восьмеро, восьмидесятый, восьмой, вот, вперед, впереди, впрямь, вроде, все, всегда, всего, все–таки, вслед, вследствие, всюду, всякий, всяко, всякое, второй, вы, выше, где, где–либо, где–нибудь, где–то, геть, глядь, гм, гоп, гы, да, да–а, да–а–а, дабы, давай, давайте, да–да, да–да–да, даже, дай, дайте, дак, данный, два, двадцатый, двадцать, двенадцатый, двенадцать, двести, двое, д–да, де, девяносто, девяностый, девятнадцатый, девятнадцать, девятый, девять, девятьсот, дель, ден, дер, десятеро, десятый, десять, ди, для, до, добро, доколе, дон, доселе, дотоле, другие, другое, другой, дудки, ды, дык, его, едва, ее, ежели, ежли, ей–богу, ей–ей, ейный, елки–палки, е–мое, если, ет, ето, ето, етот, еще, ж, же, за, заместо, зато, зачем, зачем–то, здесь, здеся, здорово, здравствуй, здравствуйте, здрасте, значит, зы, и, ибн, ибо, , идти, иже, иже, из, из–за, изнутри, изо, из–под, и–и, или, иль, именно, имхо, ин, иначе, иначе, иной, исключая, исключительно, итак, ить, их, ихний, ишь, ишь, к, ка, ка–ак, кабы, каждый, кажный, как, как– либо, как–нибудь, как–никак, како, каков, каково, каковой, какой, какой– либо, какой–нибудь, какой–никакой, какой–то, как–то, касательно, кис–кис, ко, когда, когда, когда–либо, когда–нибудь, когда–то, кое, кое–где, кое–как, кое–какой, кое–кто, кое–что, кой, кой–какой, кой–кто, кой–то, кой–что, коли, коль, конечно, который, кроме, кругом, кто, кто–кто, кто–

либо, кто–нибудь, кто–то, ку, куда, куда–либо, куда–нибудь, куда–то, куды, ку–ку, кыш, ла, ладно, ле, ли, ли, либо, лишь, лучше, ль, любой, м, мало, марш, мда, м–да, мдя, меж, между, меньше, мерси, мимо, мля, мм, м –м, ммм, м–м–м, , мочь, может, многие, многий, много, многое, много–много, мой, мол, мы, мяу, на, навроде, навсегда, над, надо, на–ка, накануне, наперекор, наподобие, напротив, насчет, нате, наш, н–да, не, неа, не–а, небось, невесть, не–е, не–е– ет, не–ет, нежели, неизвестно, некий, некогда, некого, некоторые, некоторый, некто, немало, немногие, немногий, немного, немногое, немножко, несколько, нет, нет–нет, нет–нет–нет, неужели, неужто, нехай, нечего, нечто, нешто, ни, нибудь, нигде, ниже, никак, никакой, никогда, никой, никто, никуда, ниоткуда, нипочем, нисколечко, нисколько, ниче, ниче, ничего, ничей, ничо, ничто, ничуть, ништяк, н–не, н–нет, н–ну, но, но– но, ну, ну–ка, ну–ко, ну–ну, ну–с, ну–у, нэ, о, об, оба, обо, обоего, оглы, ого, ого–го, о–го–го, один, одиннадцатый, одиннадцать, однако, одно, ой, ой–ой– ой, ок, о'кей, около, окрест, окромя, о'кэй, он, она, они, оно, оный, о–о, о–о– о, оп, остальное, остальной, остальные, от, откуда, откуда–нибудь, откуда– то, относительно, ото, отовсюду, отсюда, отсюдова, оттого, оттого–то, оттуда, оттудова, отчего, отчего–то, офф, ох, ох–хо–хо, очень, пам, пардон, первый, перед, передо, пи, пиф–паф, пли*, по, по–вашему, поверх, повсюду, по–всякому, под, поди, подле, подо, подобно, по–другому, поелику, пожалуйста, по–за, позади, по–иному, пока, покамест, покуда, полноте, полста, полтора, полтораста, полундра, помимо, по–моему, по–над, по– нашему, понеже, поперек, по–своему, посему, посередине, посередь, поскольку, после, посреди, посредине, посредством, постольку, по –твоему, потом, потому, потому–то, почем, почему, почему–либо, почему–то, почто, пошто, поэтому, поэтому–то, правда, превыше, пред, предо, прежде, при, притом, причем, про, промеж, просто, против, противу, прочая, прочее, прочий, прям, прямо, пу, пускай, пусть, путем, пущай, пшел, пятеро, пятидесятый, пятнадцатый, пятнадцать, пятый, пять, пятьдесят, пятьсот, равно, равняйсь, ради, раз, разве, ровно, р–раз, с, сам, самый, самый–самый, сверх, свое, свой, свыше, се, себе, себя, седьмой, сей, сейчас, сем, семеро, семидесятый, семнадцатый, семнадцать, семь, семьдесят, семьсот, середь,

сзади, сие, сиречь, сичас, сквозь, сколь, сколько, сколько –нибудь, сколько– то, сколь–нибудь, словно, со, собственно, согласно, сообразно, соответственно, сорок, сороковой, сорри, сотый, спасибо, спасибочки, спустя, среди, средь, сродни, становиться, сто, столь, столько, столько –то, стоп, стук, супер, супротив, сю, сюда, сюды, сяк, сякой, сям, та, та–ак, так, также, таки, тако, таков, таковой, таковский, такой, такой–сякой, такой–то, так–так, так–таки, так–так–так, так–то, там, тама, там–то, та–та, та–та–та, твой, те, тем, теперь, тик–так, типа, то, тогда, тогда–то, тож, той, тока, токмо, токо, только, только–то, топ, то–се, тот, то–то, то–то, тот–то, точно, тра–та–та, трах, третий, три, тридевять, тридцатый, тридцать, тринадцатый, тринадцать, трис, триста, трое, тсс, тс–с, ттт, туда, туда–сюда, туда–то, туды, тук, тук–тук, тук–тук–тук, тут, тута, тут–то, ту–ту, ты, тьфу, тьфу–тьфу, тьфу–тьфу–тьфу, тю, у, уа, увы, угодно, угу, уж, ужели, ужель, уй, ура, усе, у–у, ууу, у–у–у, уф, ух, фи, фон, фра, фу, ха, ха–ха, ха–ха–ха, хвать, хе, хех, хе–хе, хе–хе–хе, хи, хи–хи, хи–хи–хи, хлоп, хм, хны, хо, хорошо, хоть, хотя, хо–хо, хо–хо–хо, хошь, хр, хрясь, хто, цоб, цыц, чаво, чай, чао, че, чево, чего, чегой–то, чего–то, чей, чей–либо, чей–нибудь, чей–то, чем, через, черт–т, четверо, четвертый, че–то, четыре, четыреста, четырнадцатый, четырнадцать, чи, чик–чик, чмок, чо, чего, чрез, что, чтоб, чтобы, чтой–то, что–либо, что– нибудь, что–нить, что–о, что–о–о, что–то, что–что, чу, чур, чуть, ч–черт, ша, шалом, шестеро, шестидесятый, шестисотый, шестнадцатый, шестнадцать, шестой, шесть, шестьдесят, шестьсот, шо, шоб, што, штоб, шу, ща, щелк, э, эвон, эврика, эге, эдак, эдакий, эй, эк, эка, экий, эль, энный, эт, этак, этакий, это, этот, эт–то, эх, ээ, э–э, э–эх, эээ, э–э–э, я, яко, якобы, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, ,, ., !, №, @, #, $, л, ;, %, :, &, ?, *, (,), _, +, =, –, {, }, [,], '', ', |, \, /, a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z.

Приложение В

Список стоп–слов, дополняющий список стоп– слов

использовать, позволять, определять, осуществлять, следовать, иметь, предполагать, являться, рассматривать, рассматриваться, показывать, сформировывать, замечать, описываться, использоваться, располагать, подчеркивать, оказываться, описывать, возникать, допускать, удовлетворять, использоваться, определяться, находить, означать, приводить, составлять, называть, происходить, принимать, называться, получать, выбираться, заключаться, учитывать, вычислять, иметь, иметься, описывать, полагать, повторять, содержаться, сравниваться, находиться, обозначать, основываться, соответствовать, представлять, давать, появление, применять, применяться, требовать, интерпретировать, фиксировать, производиться, характеризовать, разрабатывать, видеть, входить, образовываться, можно, должно, подставлять, даваться, содержать, принадлежать, знать, выражаться, наличие, отсутствие, обнаружение, соответствующий, соотношение, использование, прохождение, следовательно, помощь, вычисление, действительный, например, действительно, определенный, рассмотрение, выход, ошибка, рис, схема.