


Федеральное государственное бюджетное образовательное учреждение высшего образования «Сибирский государственный аэрокосмический университет имени академика М.Ф. Решетнева»

На правах рукописи



Брестер Кристина Юрьевна

**КОЛЛЕКТИВНЫЙ ЭВОЛЮЦИОННЫЙ МЕТОД
МНОГОКРИТЕРИАЛЬНОЙ ОПТИМИЗАЦИИ В ЗАДАЧАХ АНАЛИЗА
РЕЧЕВЫХ СИГНАЛОВ**

05.13.01 – Системный анализ, управление и обработка информации
(информатика, вычислительная техника, управление)

Диссертация
на соискание ученой степени кандидата технических наук

Научный руководитель:
Семенкина Ольга Эрнестовна
доктор технических наук, профессор

Красноярск – 2016

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
Глава 1. Эволюционные алгоритмы многокритериальной оптимизации для распознавания эмоций человека по речи.....	11
1.1 Распознавание эмоций человека по речи.....	11
1.2 Обоснование интеграции эволюционных алгоритмов и средств интеллектуального анализа данных.....	14
1.3 Постановка задачи многокритериальной оптимизации.....	18
1.4 Обзор эволюционных алгоритмов многокритериальной оптимизации.....	21
ВЫВОДЫ.....	31
Глава 2. Разработка, реализация и исследование коллективного генетического алгоритма многокритериальной оптимизации.....	32
2.1 Основы функционирования генетических алгоритмов многокритериальной оптимизации.....	32
2.2 Коллективный генетический алгоритм многокритериальной оптимизации ..	36
2.3 Тестовые задачи для исследования эволюционных алгоритмов многокритериальной оптимизации.....	48
2.4 Исследование эффективности разработанного коллективного генетического алгоритма многокритериальной оптимизации.....	53
ВЫВОДЫ.....	59
Глава 3. Применение коллективного генетического алгоритма многокритериальной оптимизации для автоматического генерирования искусственных нейронных сетей в задаче распознавания эмоций по речи.....	61
3.1 Исследование эффективности классификаторов на задаче распознавания эмоций человека по речи.....	61
3.2 Искусственные нейронные сети как инструмент интеллектуального анализа данных.....	69

3.3	Автоматическое генерирование полносвязных перцептронов многокритериальным генетическим алгоритмом	77
3.4	Исследование эффективности многокритериального генетического алгоритма автоматического проектирования полносвязных перцептронов	82
	ВЫВОДЫ	92
	Глава 4. Извлечение информативных признаков в задаче распознавания эмоций человека по речи коллективным генетическим алгоритмом многокритериальной оптимизации.....	93
4.1	Основные подходы к извлечению информативных признаков.....	93
4.2	Исследование эффективности подхода filter на задаче распознавания эмоций человека по речи.....	98
4.3	Исследование эффективности подхода wrapper на задаче распознавания эмоций человека по речи	110
4.4	Сравнительный анализ подходов filter и wrapper на задаче распознавания эмоций человека по речи	116
	ВЫВОДЫ	119
	ЗАКЛЮЧЕНИЕ	120
	СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	121
	Приложение 1. Сравнение эффективности разработанного алгоритма с победителями соревнования SEC2009	139

ВВЕДЕНИЕ

Актуальность. На сегодняшний день качество распознавания устной речи интеллектуальными диалоговыми системами стремительно повышается. Однако для раскрытия смысла сообщения не достаточно знать совокупность слов, составляющих речевой сигнал. Правильная интерпретация контекста невозможна без учета невербальных сигналов, сопровождающих словесные конструкции. Поэтому распознавание эмоционального состояния говорящего является ключевым аспектом в ходе анализа устной речи. Однако в связи с особенностями задачи применение стандартных методов интеллектуального анализа данных не обеспечивает высокой эффективности. В качестве альтернативы предлагается использовать подходы, основанные на эвристических процедурах.

Дисциплина, зародившаяся на стыке машинного обучения (*англ.* machine learning) и эволюционных вычислений, переросла в направление (*англ.* evolutionary machine learning), охватывающее не только методы извлечения знаний из баз данных, проектирования моделей, алгоритмы их обучения, но и всю совокупность средств эволюционного поиска, необходимых для тонкой настройки данных технологий. В большинстве случаев применение эволюционных методов в области машинного обучения ограничивается привлечением стохастических алгоритмов для решения задач однокритериальной оптимизации. Однако учет нескольких критериев качества позволяет расширить возможности применяемого алгоритмического аппарата. Формальную модель такого рода задач целесообразно представлять как совокупность функционалов, отражающих те или иные аспекты проблемы: они могут быть согласованными, независимыми или конкурирующими.

За последние тридцать лет было предложено немало эффективных эволюционных, а именно генетических алгоритмов (ГА) многокритериальной оптимизации, реализующих принцип Парето-доминирования, идею элитизма, модификации селективного отбора и т.д. На сегодняшний день эволюционные методы широко используются не только как мощный алгоритмический аппарат

для решения оптимизационных задач, но и как эффективный вспомогательный инструментарий в области интеллектуального анализа данных.

Поэтому разработка и исследование эволюционных методов многокритериальной оптимизации, демонстрирующих высокую эффективность при их использовании в области машинного обучения, в том числе в сфере анализа речевых сигналов, является **актуальной научно-технической задачей**.

Несмотря на универсальность эволюционных алгоритмов, их довольно высокую эффективность в смысле точности и надежности и возможность функционирования в динамически изменяющемся пространстве, некоторые ученые приводят аргументы против интеграции машинного обучения и эволюционных вычислений. Во-первых, они апеллируют к необходимости выбора нужного алгоритма из широкого спектра эвристических методов, эффективность которых существенно меняется при переходе от одной задачи к другой. Во-вторых, по сравнению со стандартными аналогами эволюционные алгоритмы требуют больше вычислительных ресурсов. Поэтому **целью диссертационной работы** является повышение эффективности эволюционных методов, используемых для решения задач многокритериальной оптимизации в области анализа речевых сигналов.

Открытые вопросы рассматриваемой предметной области и сформулированная цель работы предопределили совокупность решаемых **задач**:

1. Проанализировать целесообразность применения эволюционных методов в задаче распознавания эмоций человека по речи.
2. Реализовать и исследовать ряд стандартных генетических алгоритмов многокритериальной оптимизации.
3. Спроектировать островную модель параллельного генетического алгоритма многокритериальной оптимизации, компонентами которого являются методы, основанные на различных эвристиках.
4. Реализовать предложенную схему алгоритма. Исследовать эффективность данного подхода на тестовых задачах.

5. Автоматизировать процесс генерирования структуры искусственной нейронной сети с учетом дополнительного критерия «вычислительная сложность модели».

6. Разработать многокритериальную математическую модель задачи извлечения информативных признаков из баз данных. На основе предложенной модели реализовать процедуру отбора релевантных атрибутов при помощи разработанного генетического алгоритма.

7. Апробировать предложенный алгоритмический аппарат и реализованное программное обеспечение на практических задачах распознавания эмоций человека по речи.

Методы исследования. В данной работе использовались методы эволюционных вычислений, нейросетевого моделирования, оптимизации, теории вероятности и математической статистики, системного анализа и другие.

Научная новизна результатов диссертационной работы состоит в следующем:

1. Разработан новый коллективный параллельный генетический алгоритм для решения задач многокритериальной оптимизации, основанный на островной модели кооперации и отличающийся от своих компонент параллельной структурой, а также сочетанием различных концепций эволюционного поиска.

2. Разработан новый метод проектирования архитектуры нейросетевых классификаторов, позволяющий генерировать набор альтернативных моделей различной точности и вычислительной сложности, отличающийся от известных подходов возможностью выбора нейронной сети с компактной структурой, соответствующей заданному уровню точности.

3. Разработаны новые эволюционные методы извлечения информативных признаков из баз данных при решении классификационных задач, основанные на двухкритериальных оптимизационных моделях и отличающиеся от известных подходов способом представления решения и возможностью формирования коллективов классификаторов.

4. Реализованные алгоритмические схемы были впервые применены для решения задачи распознавания эмоций человека по речи. Использование разработанных технологий позволило повысить качество получаемых решений.

Теоретическая значимость результатов диссертационного исследования состоит в разработке нового многокритериального эволюционного алгоритма, сочетающего в себе три метода, основанных на различных концепциях. Островная модель алгоритма позволяет избежать выбора конечным пользователем наиболее эффективного метода из числа имеющихся, а параллельная реализация приводит к сокращению временных затрат, требуемых для работы алгоритма, что имеет существенное значение при использовании эволюционного поиска в области машинного обучения.

Практическая ценность. Разработанные алгоритмические схемы отличаются ориентацией на конечного пользователя, поскольку не требуют выбора оптимизационной процедуры при решении задач предобработки данных (отбора информативных признаков) и классификации. Параллельная работа компонент эволюционного алгоритма позволяет экономить время, требуемое для решения задачи.

Реализованные подходы были успешно применены для решения задачи распознавания эмоций человека по речи, что является одним из ключевых направлений интеллектуализации диалоговых систем.

Реализация результатов работы. Разработанные алгоритмы были использованы при выполнении исследований в рамках проекта «Разработка методов и технологий распараллеливания алгоритмов интеллектуального анализа данных в многопроцессорных и распределенных вычислительных системах» (ГК 14.В37.21.1521, в рамках ФЦП); российско-германских проектов, посвященных разработке методов интеллектуального анализа данных в мультилингвистических системах, выполняемых в рамках ФЦП по ГК № 11.519.11.4002 и 16.740.11.0742; проекта «Модели и алгоритмы функционирования систем интеллектуального анализа данных с пространственно-временным механизмом адаптации для

решения задачи моделирования и оптимизации сложных технических систем» (грант Президента РФ 2014-2015 гг. МК-5391.2014.9) и проекта № 140/14 «Разработка теоретических основ эволюционного проектирования интеллектуальных информационных технологий анализа данных» (государственное задание, утвержденное Минобрнауки России на 2014–2016 гг.). Проводимые исследования также были поддержаны Фондом содействия развитию малых форм предприятий в научно-технической сфере в рамках программы «Участник молодежного научно-инновационного конкурса» (У.М.Н.И.К.), 2014–2015 гг.

Семь программных систем зарегистрированы в Роспатенте. Разработанное программное обеспечение используется в учебном процессе Института информатики и телекоммуникаций СибГАУ при выполнении лабораторных и курсовых работ.

Программные системы переданы для использования в две инновационные IT-компании г. Красноярска, а также в Ульмский университет (г. Ульм, Германия) для проведения исследования в сфере распознавания персональных характеристик пользователей диалоговых систем.

Основные защищаемые положения:

1. Разработанный параллельный эволюционный алгоритм многокритериальной оптимизации превосходит свои компоненты, включенные в состав островной модели, по точности, надежности и оперативности работы.
2. Разработанный подход к проектированию нейросетевых классификаторов позволяет строить системы, эффективные по точности и вычислительной сложности.
3. На множестве представленных задач применение эволюционных методов извлечения информативных признаков, основанных на двухкритериальных оптимизационных моделях, позволяет повысить точность классификации.
4. Спроектированные схемы извлечения информативных признаков являются эффективным средством предобработки данных и могут быть

успешно использованы в сочетании с коллективом классификаторов различной природы.

Апробация работы. Результаты диссертационной работы докладывались в период 2010-2015 гг. более чем на 20 конференциях различного уровня, среди которых: INTERSPEECH 2015 (Dresden, Germany, 2015), Sixth International Conference on Swarm Intelligence (Beijing, China, 2015), IEEE Congress on Evolutionary Computation (Sendai, Japan, 2015), Informatics in Control, Automation and Robotics (Colmar, France, 2015; Vienna, Austria, 2014), International Conference on Engineering and Applied Sciences Optimization (Kos Island, Greece, 2014), The 9th edition of the Language Resources and Evaluation Conference (Reykjavik, Iceland, 2014), International Conference on Computer Science and Artificial Intelligence (Wuhan, China, 2014), International Workshop on Mathematical Models and its Applications (Krasnoyarsk, Russia, 2013, 2014, 2015), Пятая Международная конференция «Системный анализ и информационные технологии» (Красноярск, 2013), Третья Всероссийская научная конференция с международным участием «Теория и практика системного анализа» (Рыбинск, 2014), Всероссийская научно-техническая конференция «Информационно-телекоммуникационные системы и технологии» (Кемерово, 2014), XIII Международная научная конференция «Молодежь. Общество. Современная наука, техника и инновации» (Красноярск, 2014), XIII Международная научная конференция «Интеллект и наука» (Железногорск, 2013), XVI и XVIII Международные научные конференции «Решетневские чтения» (Красноярск, 2012, 2014), и др. Кроме того, отдельные результаты работы были доложены на научных семинарах института информационных технологий университета г. Ульм (Германия, 2014), Мариборского университета в г. Крань (Словения, 2014), университета Восточной Финляндии в г. Куопио (Финляндия, 2015). Диссертация в целом обсуждалась на научно-технических семинарах кафедры системного анализа и исследования операций СибГАУ и кафедры систем автоматизированного проектирования (РК6) НИУ МГТУ им. Н.Э. Баумана.

Публикации. По материалам данной работы опубликовано 26 печатных работ, в том числе 6 статей в научных изданиях Перечня ВАК, 6 - в изданиях, индексируемых в международной базе Scopus, 2 – в изданиях, индексируемых в Web of Science.

Структура работы. Работа состоит из введения, четырех глав, заключения, списка литературы и приложений.

Глава 1. Эволюционные алгоритмы многокритериальной оптимизации для распознавания эмоций человека по речи

1.1 Распознавание эмоций человека по речи

На сегодняшний день качество распознавания устной речи интеллектуальными диалоговыми системами стремительно повышается. Однако для раскрытия смысла сообщения не достаточно знать совокупность слов, составляющих речевой сигнал. Наряду с основным вербальным каналом человек зачастую используют паралингвистические средства общения, например, фонационные (тембр речи, ее темп и громкость) или кинетические (жесты, мимика). Правильная интерпретация контекста невозможна без учета невербальных сигналов, сопровождающих словесные конструкции [54]. Поэтому распознавание эмоционального состояния говорящего является ключевым аспектом в ходе анализа устной речи [125].

Научное направление, целью которого является разработка и создание систем, способных распознавать, интерпретировать, обрабатывать, а также имитировать эмоции человека, называется *аффективные вычисления* (англ. *affective* – эмоциональный). Это междисциплинарная область знаний, включающая в себя технологии искусственного интеллекта, психологии и когнитивистики.

Согласно Picard R. (Массачусетский технологический институт (MIT)) [100, 101], основательнице современных аффективных вычислений, в общем случае для распознавания психоэмоционального состояния человека могут быть использованы как явные, так и менее очевидные характеристики. К явным относят выражение лица, позу, интонационные особенности голоса, к менее очевидным – сердечный ритм, частоту дыхания, артериальное давление. Понятно, что в повседневной жизни аффективные интерфейсы, используемые в ходе «человеко-машинной» коммуникации, лишены возможности измерять весь спектр перечисленных признаков с помощью сенсоров. Зачастую анализ эмоционального

состояния человека осуществляется на основе визуальной информации или акустических характеристик голоса. Разрабатываются также и мультимодальные системы, использующие как видео-, так и аудиоканал для получения измерений. В данной работе рассматривается подход к распознаванию эмоционального состояния человека по совокупности акустических характеристик речевого сигнала.

Еще в 1994 г. ученым Damasio были выделены две группы эмоций – первичные и вторичные [45]. К первой относятся такие примитивные эмоции как грусть, страх, печаль, радость, ко второй – стыд, вина, зависть, то есть эмоции, возникающие в ходе социального взаимодействия. На сегодняшний день учеными предприняты попытки автоматического распознавания как первичных, так и вторичных эмоций по речи человека. В работе [133] Zhang исследует механизмы определения таких состояний, как уверенность, растерянность и разочарованность. Kwon разработал подход для идентификации стресса [89]. Steidl представил метод распознавания сочувствия [121]. Попытки выявить состояние человека, говорящего неправду, предпринимали Hirschberg [71] и Graciarena [64]. Batliner разработал алгоритмический аппарат для определения волнения [25]. В работе [24] Ang исследует подходы для идентификации раздражения и разочарования по речи. Кроме того, особую группу представляют проекты, направленные на выявление аутизма и его лечение у детей [95, 116].

Общая схема процесса распознавания эмоций включает в себя несколько этапов. Во-первых, необходимо построить базу данных, используемую для обучения классификационной модели. Поскольку аффективные вычисления являются развивающейся дисциплиной, то формирование речевых корпусов остается актуальной задачей. Существует два основных метода построения баз данных. Согласно первому, голосовые записи заимствуются из различных телевизионных шоу, интервью или радиоэфиров, а затем эксперты оценивают эмоциональную окраску высказываний. Во втором случае привлекаются актеры для озвучивания каких-либо фраз, диалогов или текстов, имеющих определенное эмоциональное содержание (злость, счастье или нейтральное). Однако оба метода

требуют значительных временных затрат как на работу экспертов, так и на озвучивание выражений.

Далее из каждой сохраненной звукозаписи необходимо извлечь признаки, описывающие особенности сигнала [14]. В рамках конференции INTERSPEECH 2009 был предложен репрезентативный набор акустических характеристик, включающий в себя 384 признака [117, 119]: максимальное, минимальное, среднее значения или среднеквадратическое отклонение измерений, описывающих речевой сигнал, его высоту, вибрации, интенсивность и т.п. Широкое распространение получили программные системы OpenSMILE (Мюнхенский технический университет) [55] и Praat (Амстердамский университет) [27], ориентированные на извлечение указанных признаков.

Затем полученные числовые измерения используются для проектирования классификационной модели и ее настройки (обучение с учителем). Из звукозаписи, подлежащей анализу, также извлекаются акустические характеристики и подаются на вход обученной модели для получения оценки принадлежности эмоционального состояния говорящего к одному из классов, представленных в обучающей выборке.

Несмотря на то, что уже сейчас удастся распознавать эмоции человека по речи с достаточно высокой точностью, остается ряд открытых вопросов:

- использование всего набора акустических характеристик затрудняет работу обучающего алгоритма при настройке классификатора, поэтому целесообразно осуществлять отбор атрибутов, релевантных для рассматриваемой задачи (предлагаемый подход и полученные результаты опубликованы в [35]);
- невозможно предсказать заранее, какой из классификаторов окажется наиболее эффективным для конкретной базы данных, поэтому возможна разработка методов, основанных на привлечении ансамблей различных моделей (результаты данного исследования опубликованы в [32, 33]), а также подходов для автоматического проектирования структуры классификатора под конкретную задачу (опубликовано в [30]);

– в связи с особенностями обучающей выборки (высокая размерность вектора признаков, значительное количество обучающих примеров) применение стандартных методов интеллектуального анализа данных не обеспечивает высокой эффективности. Было установлено [35], что использование метода главных компонент приводит к снижению точности распознавания эмоций, а в результате проектирования классификаторов, например, нейронных сетей, по общепринятым технологиям структура модели получается избыточной.

Поэтому в диссертационной работе в качестве альтернативы стандартным методам машинного обучения, применяемым к задаче распознавания эмоций человека по речи, рассматриваются подходы, основанные на эвристических процедурах. Далее представлены общие сведения о средствах интеллектуального анализа данных, а также освещены положительные и отрицательные эффекты внедрения стохастических методов в аппарат машинного обучения.

1.2 Обоснование интеграции эволюционных алгоритмов и средств интеллектуального анализа данных

В 1989 г. Григорием Пятецким-Шапиро были предложены два термина «*data mining*» и «*knowledge discovery in data*», которые в настоящее время являются ключевыми понятиями интеллектуального анализа данных [99]. В переводе с английского «*data mining*» означает «просев информации, добыча или извлечение данных», однако более полным и точным является словосочетание «*knowledge discovery in data*», интерпретируемое как «обнаружение знаний в базах данных».

На сегодняшний день существует две точки зрения на то, как соотносятся представленные понятия [69]:

- введенные термины синонимичны и равноправны;
- «*data mining*» является главным этапом в процессе «*knowledge discovery*».

В связи с тем, что вторая точка зрения представляется более обоснованной, будем придерживаться ее в данной работе и рассмотрим выбранную позицию детально.

В общем случае процесс обнаружения знаний в базах данных («knowledge discovery») включает в себя следующие этапы [61]:

- 1) интеграция данных (*англ.* data integration) – эта стадия актуальна в том случае, если данные поступают из разных источников. Возникает необходимость в обнаружении несоответствий в названиях атрибутов или градаций (имен) номинальных переменных;
- 2) очистка данных (*англ.* data cleaning) – заполнение пропусков данных, обнаружение выбросов, ошибок и зашумленных измерений;
- 3) отбор признаков (*англ.* data selection) – выбор атрибутов, релевантных для рассматриваемой задачи;
- 4) трансформация данных (*англ.* data transformation) – применение операции агрегирования (линейное или нелинейное преобразование) для сокращения числа атрибутов;
- 5) извлечение данных (*англ.* data mining) – ключевой этап проектирования и применения интеллектуальных технологий для обнаружения шаблонов и закономерностей в базе данных;
- 6) оценка выявленных шаблонов (*англ.* pattern evaluation) – обнаружение новых, ранее неизвестных, интерпретируемых закономерностей, представляющих интерес и практическую ценность;
- 7) представление знаний (*англ.* knowledge presentation) – представление полученных закономерностей в визуальной и иной форме пользователю.

В последовательности указанных этапов условно выделяют три стадии: предобработка данных, извлечение знаний (или data mining) и постобработка полученных закономерностей (рисунок 1.1).

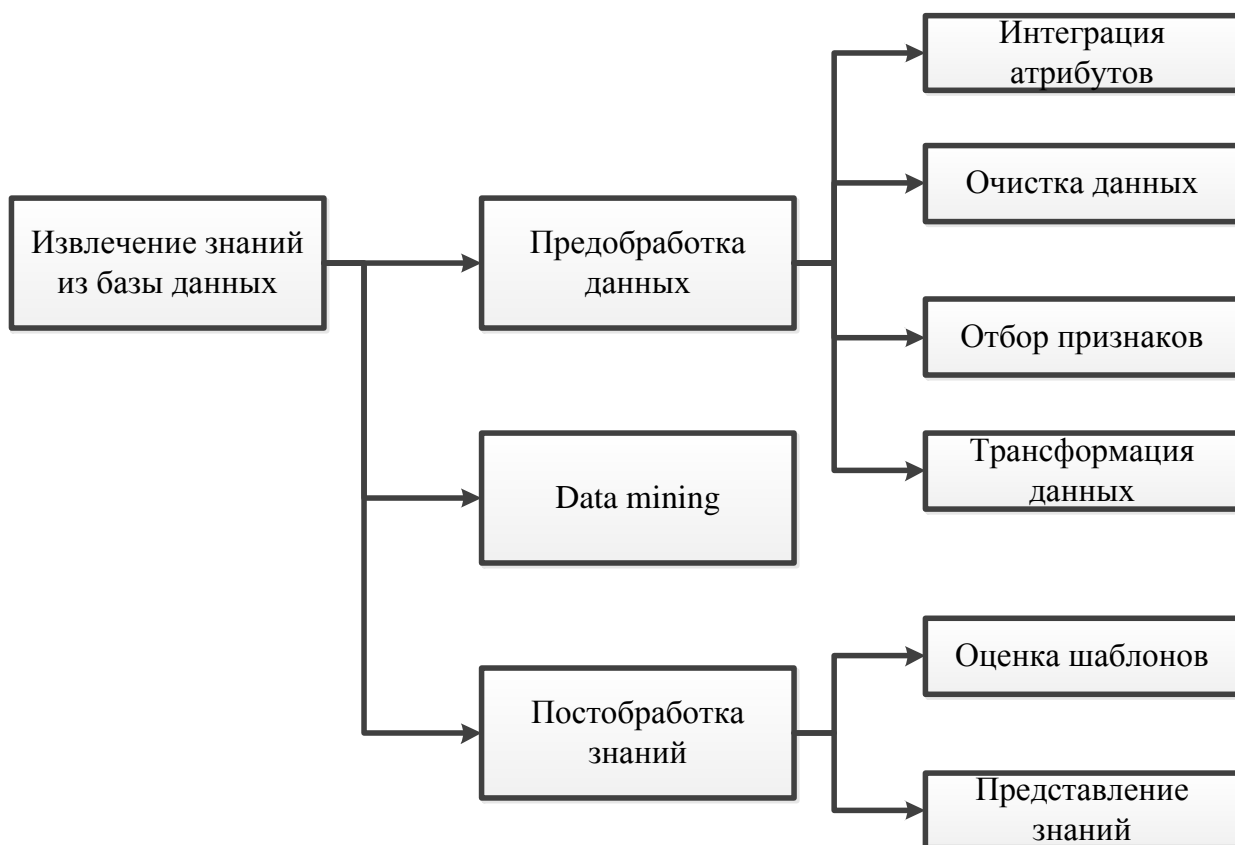


Рисунок 1.1 – Стадии процесса извлечения знаний из базы данных

Моделирование найденных закономерностей осуществляется в рамках машинного обучения (*англ.* machine learning) [96]. Причем конечной целью является не просто точное воспроизведение выявленных шаблонов, а их обобщение с целью анализа новых объектов, не представленных в базе данных. Другими словами, машинное обучение – это область искусственного интеллекта, включающая в себя методы проектирования моделей, а также алгоритмы их обучения, предназначенные для решения задач классификации, кластеризации, аппроксимации, снижения размерности и т.п. (рисунок 1.2).

В последнее время все большую популярность набирает дисциплина, зародившаяся на стыке машинного обучения и эволюционных алгоритмов (*англ.* genetics-based machine learning) [86]. Ярким примером является применение генетических алгоритмов для обучения нейронных сетей или генерирования базы правил для нечетких классификаторов.

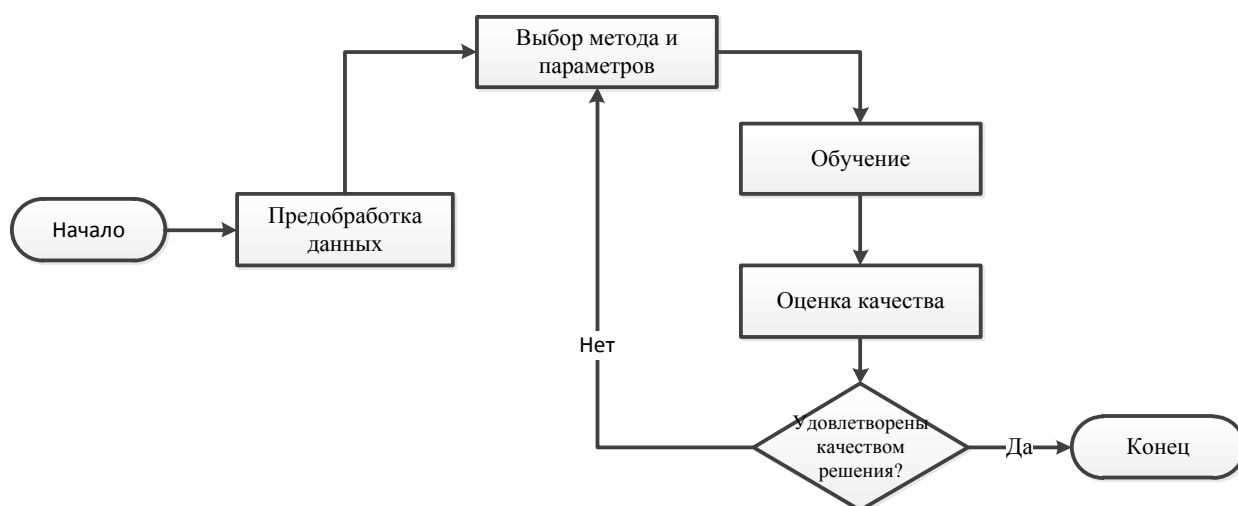


Рисунок 1.2 – Обобщенная модель процесса машинного обучения

Можно выделить ряд факторов, обуславливающих растущий интерес к применению эволюционных методов в области машинного обучения [59]:

- 1) точность моделей, обученных с привлечением эволюционного поиска, сравнима с точностью моделей, обученных при помощи стандартных подходов. Кроме того, гибридизация глобального и локального поиска позволяет получать еще более эффективные решения;
- 2) применение эволюционных алгоритмов позволяет встраивать отбор информативных признаков в процесс обучения модели;
- 3) эволюционные методы пригодны для работы в динамически изменяющейся среде;
- 4) данные алгоритмы легко подвергаются распараллеливанию;
- 5) эволюционные методы отличаются универсальностью.

Однако учеными также приводятся аргументы, ставящие под сомнение целесообразность применения эволюционного поиска в области машинного обучения. В большинстве случаев стохастические эволюционные методы работают дольше, чем их стандартные аналоги. Но с развитием аппаратной части ЭВМ, появлением кластеров и расширением вычислительных мощностей данный аргумент становится все менее убедительным. Кроме того, возможность распараллеливания вычислений также означает сокращение временных затрат. В качестве еще одного аргумента исследователи приводят тот факт, что различные

эволюционные алгоритмы не демонстрируют одинаковой эффективности для спектра задач. Для решения данной проблемы учеными активно разрабатываются адаптивные версии алгоритмов с автоматическим выбором вариантов генетических операторов [19], а также кооперативно-конкурирующие модификации, работа которых основана на привлечении сразу нескольких алгоритмов, реализующих различные эвристики [18].

С учетом указанных факторов в рамках данной работы был спроектирован и исследован коллективный параллельный генетический алгоритм (ГА) для решения задач многокритериальной оптимизации, в основу функционирования которого была положена островная модель (глава 2). В большинстве случаев применение эволюционных методов в области машинного обучения ограничивается привлечением стохастических алгоритмов для решения задач однокритериальной оптимизации. Однако учет нескольких критериев качества позволяет расширить возможности применяемого алгоритмического аппарата.

Далее рассмотрим постановку задачи многокритериальной оптимизации, а также историю развития эволюционных алгоритмов, оперирующих несколькими критериями качества.

1.3 Постановка задачи многокритериальной оптимизации

В задачах оптимизации целевой функционал, называемый также критерием качества, представляет собой количественную модель цели функционирования объекта. При этом на практике редко удается выразить цель одним критерием.

Уже в 50-х годах XX века сообщество по исследованию операций начало разрабатывать подходы для решения многокритериальных задач. Типичным методом являлось приведение математической модели с несколькими целями к однокритериальной постановке с помощью таких подходов как: введение агрегированного критерия или взвешенной метрики, целевое программирование (взвешенное, лексикографическое, минимаксное) и т.д. Однако данные подходы имеют следующие ограничения: во-первых, настройка параметров (весовых

коэффициентов) требует дополнительных знаний о проблеме; во-вторых, было установлено [129], что эти методы чувствительны к особенностям задачи: так, например, использование аддитивной свертки неэффективно в случае, если решения в критериальном пространстве образуют вогнутую поверхность. Кроме того, искомым результатом многокритериальной оптимизационной задачи может быть не одна, а сразу несколько несравнимых между собой точек, что приводит к необходимости совершения многократных запусков алгоритма с различными настройками для получения множества компромиссных точек.

Осознание недостатков существующих методов и того факта, что зачастую на практике введенные критерии качества являются конфликтующими, привело к необходимости пересмотра понятия «оптимальность». На тот момент в экономике уже было предложено толкование оптимальности по Парето, которое и было заимствовано математиками для решения многокритериальных задач. Первоначально данная концепция была предложена Фрэнсисом Исидро Эджуортом (Edgeworth F.Y., ирландский экономист, 1845 –1926 гг.) [53], а позже обобщена Вильфредо Парето (Pareto V., итальянский инженер, экономист и социолог, 1848–1923 гг.) [98]. Суть принципа Парето-оптимальности состоит в том, что улучшение частного критерия не может быть достигнуто без ухудшения остальных показателей эффективности. Таким образом, в основу большинства алгоритмов многокритериальной оптимизации легла идея поиска множества компромиссных точек в допустимой области.

Пусть функционирование системы оценивается по m критериям качества f_1, f_2, \dots, f_m [22]:

$$F(\bar{x}) = \{f_1(\bar{x}), f_2(\bar{x}), \dots, f_m(\bar{x})\} \rightarrow \text{extr}, \quad (1.1)$$

где $\bar{x} \in S$, S - область допустимых решений, $f_i: R^n \rightarrow R$, $i = \overline{1, k}$ - целевые функции. Они могут быть нейтральными, согласованными или противоречивыми. Если функции f_1, f_2, \dots, f_m имеют оптимум в одной и той же точке $\bar{x}^* \in S$, то говорят, что задача (1.1) имеет идеальное решение. Случаи существования идеального решения в многокритериальной задаче крайне редки. Поэтому основная проблема

при рассмотрении задачи – формализация принципа оптимальности, т.е. определение того, в каком смысле одно «оптимальное» решение лучше других. Наибольший интерес представляют случаи именно с конфликтующими критериями, поскольку улучшение одного из них приводит к ухудшению других. В данном случае решение ищется на основе компромисса, суть которого заключается в идее Парето-доминирования.

Решение \bar{x}^* называется недоминируемым (паретовским), если во множестве допустимых альтернатив не существует решения, которое по целевым функциям было бы не хуже, чем \bar{x}^* , и, по крайней мере, по одной целевой функции было бы строго лучше, чем \bar{x}^* . Недоминируемые точки из области определения составляют множество Парето, их образ в пространстве критериев – фронт Парето. Решение многокритериальной задачи целесообразно выбирать из множества Парето, эти точки не могут быть предпочтены друг другу, поэтому после формирования данного множества задача считается решенной.

Рисунок 1.3 иллюстрирует отношение Парето-доминирования в двухкритериальном пространстве. Пусть $f_1 \rightarrow \min$ и $f_2 \rightarrow \min$, тогда справедливы отношения $A < B$ (B доминирует A), $A < C$ (C доминирует A), $B \approx C$ (B и C равноценны, недоминируемы по Парето).

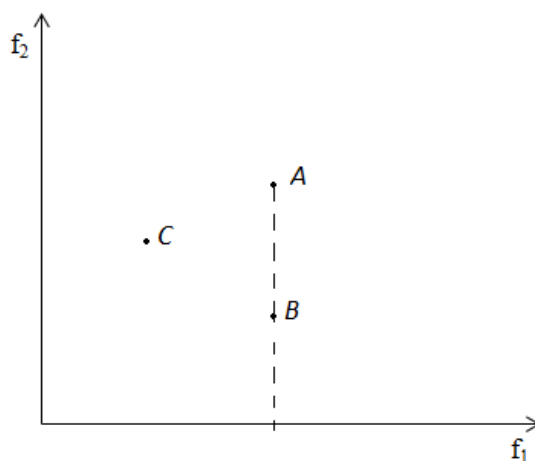


Рисунок 1.3 – Иллюстрация отношения Парето-доминирования в двухкритериальном пространстве

Если применение стандартных методов теории оптимизации становится невозможным в силу свойств целевых критериев и ограничений, то предпочтение отдается эволюционным (а именно генетическим) алгоритмам.

Первое применение эволюционных методов для решения задач многокритериальной оптимизации относится к середине 80-х гг. XX века [112]. Данные алгоритмы оперируют сразу многими точками-кандидатами, что является основной предпосылкой их использования для определения репрезентативной аппроксимации множества Парето. Кроме того, данные методы способны работать как с непрерывным, так и с дискретным поисковым пространством, что также представляется их неотъемлемым преимуществом.

1.4 Обзор эволюционных алгоритмов многокритериальной оптимизации

Принято считать, что первым эволюционным алгоритмом многокритериальной оптимизации (*англ.* Multi-Objective Evolutionary Algorithm, *сокр.* MOEA) был метод *Vector Evaluated Genetic Algorithm (VEGA)* (Schaffer D., 1985 г.) [112]. В основу метода *VEGA* легла алгоритмическая схема стандартного ГА с модифицированным оператором селекции. На каждом поколении формируется несколько подмножеств индивидов-родителей, которые участвуют в воспроизводстве потомков. Каждая группа генерируется с помощью стандартного оператора пропорциональной селекции с учетом пригодности индивидов по какому-то одному конкретному критерию. Далее отобранные индивиды объединяются вместе для получения популяции потомков с использованием операторов скрещивания и мутации.

Основная проблема данного метода заключается в отсутствии механизма, позволяющего сохранять недоминируемые по Парето решения, поскольку селективный отбор ориентирован на индивидов с высокой пригодностью по одному фиксированному критерию.

В 1989 г. Голдберг (Goldberg) в своей работе [62] предложил две основные фундаментальные концепции проектирования MOEA, чем вызвал

дополнительный интерес среди исследователей данной области. Во-первых, идея Парето-доминирования упоминалась как основной принцип работы оператора селекции. Кроме того, было введено понятие «ниша», использующееся в качестве ключевого инструмента поддержания разнообразия решений, составляющих аппроксимацию фронта Парето.

В 1993 г. Fonseca и Fleming представили первый MOEA [56], основанный на принципе Парето-доминирования. Это был эвристический метод, оперирующий бинарными строками, т.е. генетический алгоритм многокритериальной оптимизации (*англ.* Multi-Objective Genetic Algorithm, *сокр.* MOGA) [39].

После этого было разработано множество MOEA, использующих принципы введенные Голдбергом. Среди наиболее эффективных методов того времени выделяют *Non-dominated Sorting Genetic Algorithm (NSGA)* (Srinivas и Deb, 1994 г.) [120] и *Niched-Pareto Genetic Algorithm (NPGA)* (Horn, Nafpliotis, Goldberg, 1994 г.) [74].

В 1999 г. Zitzler и Thiele [141] представили рейтинг предложенных ранее методов, согласно которому алгоритм MOEA (Fonseca и Fleming, 1993 г.) [56] оказался наиболее эффективным. Далее следовали NPGA, NSGA и VEGA.

В дополнение к упомянутым ранее методам можно отметить подход, разработанный в 1992 г. исследователями Tanaka и Tanino [122]. Они впервые предложили внедрить в MOEA механизм учета предпочтений лица, принимающего решение (ЛПР). Более того в 1998 г. Fonseca и Fleming представили MOEA, в котором предпочтения ЛПР использовались алгоритмом в интерактивном режиме [57], что в свою очередь, позволяло исследовать области пространства поиска, наиболее интересующие пользователя.

На данном этапе эффективность алгоритмов исследовалась на задачах с двумя критериями и зачастую оценивалась лишь визуально. Однако в 1996 г. в работе ученых Fonseca и Fleming была введена первая метрика, оценивающая полученную алгоритмом аппроксимацию [58].

Следующим шагом в развитии MOEA стало использование стратегии элитизма. В методе *Strength Pareto Evolutionary Algorithm (SPEA)* (Zitzler и Thiele,

1999 г.) [140] недоминируемые решения, найденные в ходе работы алгоритма, сохраняются в так называемое архивное множество. После появления работ об эффективности метода *SPEA* исследователи стали активно внедрять стратегию элитизма в разрабатываемые алгоритмы. Данная концепция стала третьим основным принципом при проектировании МОЕА, наряду с Парето-доминированием и стремлением поддерживать разнообразие во множестве найденных решений [40].

В период с 1999 г. по 2003 г. основные направления в исследовании МОЕА были сосредоточены на улучшении существующих алгоритмов. В 2002 г. была представлена усовершенствованная версия *SPEA2* [139], в которой процедуры оценки пригодности и кластеризации были модернизированы с целью поддержания наибольшего разнообразия кандидатов решений в критериальном пространстве. Кроме того, были предложены метод *Pareto Envelope-based Selection Algorithm (PESA)* [44] и его улучшенная версия *PESA-II* [42], а также *Non-dominated Sorting Genetic Algorithm II (NSGA-II)* [49].

Согласно исследованию [129], метод *NSGA-II* – наиболее распространенный из всех МОЕА: обнаружено 9767 цитирований на момент 09.10.2013. Рассмотрим его принцип работы более подробно.

В данном алгоритме, как и в методах *SPEA* и *SPEA2*, реализована идея элитизма, однако в иной алгоритмической форме. *NSGA-II* не предполагает наличие внешнего архивного множества, зато на стадии селективного отбора популяции родителей и потомков объединяются вместе с целью формирования множества индивидов, которые будут участвовать в воспроизводстве. Согласно [22], *NSGA-II* более эффективен, чем его предшественники и с точки зрения вычислительных ресурсов, и в смысле качества получаемых решений. Однако его эффективность резко снижается при увеличении числа целевых критериев.

Таким образом, за данный период МОЕА претерпели существенное развитие. Были разработаны основополагающие принципы для проектирования алгоритмических схем МОЕА, среди которых:

- использование отношения Парето-доминирования для выявления более пригодных индивидов;
- поддержание разнообразия среди точек-кандидатов для получения репрезентативной аппроксимации фронта Парето;
- применение стратегии элитизма для сохранения наилучших решений.

МОЕА начинают широко применяться во многих прикладных областях, таких как электротехника, авиационная инженерия, робототехника, планирование и т.д. [47] Однако вместе с ростом интереса к МОЕА было выявлено, что методы, основанные на Парето-доминировании, например, *MOGA*, *NSGA-II* и *SPEA2*, сталкиваются со сложностями при решении многокритериальных задач с тремя и более критериями. Purshouse и Fleming в 2003 г. первыми выявили эту проблему и выяснили, что поисковая способность подходов, основанных на Парето-доминировании, существенно ухудшается при увеличении числа целей [103].

Задачи, оперирующие тремя и более критериями, возникают во многих приложениях, таким образом, разработка эффективных алгоритмов для их решения стала актуальной задачей. Кроме того, исследователи осознали, что часто ЛПР не заинтересован в получении аппроксимации фронта Парето целиком, в то время как особый интерес могут представлять отдельные его части. Поэтому было предложено несколько *MOEA*, основанных на интеграции предпочтений ЛПР в работу алгоритма.

В 2008 г. Ishibuchi и Tsukamoto [79] представили доклад, посвященный обзору проблем, с которыми приходится сталкиваться МОЕА при решении оптимизационных задач с большим количеством целевых критериев.

1. Отношение Парето-доминирования теряет свою эффективность при сравнении альтернатив в критериальном пространстве высокой размерности. Большинство точек популяции становится недоминируемыми, если число критериев велико. Доля недоминируемых векторов во множестве случайно сгенерированных решений возрастает экспоненциально с числом целей. Вследствие чего производится недостаточное селективное давление для направления поиска к Парето-

оптимальному фронту. Это приводит к низкой эффективности алгоритмов таких, как *MOGA*, *NSGA-II* и *SPEA2*, основанных на идее Парето-доминирования [76]. Более того, МОЕА становятся более чувствительными к выбору параметров [104].

2. Количество точек, необходимых для построения репрезентативной аппроксимации фронта Парето, возрастает экспоненциально с числом критериев. Это может вызвать трудность для ЛПП при выборе наиболее предпочтительного решения из множества представленных кандидатов.

3. Возникают сложности с визуальным представлением полученного аппроксимирующего множества (в отличие от случая двух- или трехкритериальной задачи).

Решением первой проблемы может быть использование модифицированных отношений Парето-доминирования, о которых речь пойдет далее.

Что касается второй трудности, то возможно внедрение предпочтений ЛПП как априори, так и в процессе работы МОЕА, для направления поиска в регионы, представляющие для пользователя особый интерес.

Для визуального представления аппроксимирующего множества предложены методы, использующие параллельные координаты [77], графики рассеяния [124], т.д. Кроме того, полезно применять к результирующему множеству методы снижения размерности, чтобы поместить целевые векторы в пространство меньших измерений.

Для решения первой проблемы было предложено несколько методов, которые можно классифицировать на категории [43]: модификации алгоритмов, основанных на идее Парето-доминирования; алгоритмы, основанные на методах ранжирования; алгоритмы, использующие методы сокращения числа критериев; алгоритмы, основанные на декомпозиции.

Но остается открытым вопрос: какое количество недоминируемых решений, найденных алгоритмом, следует сохранять? С математической точки зрения считается, что для получения репрезентативного множества целесообразным является сохранение всех недоминируемых решений в архивном множестве.

Однако на практике данная цель оказывается трудно достижимой, что вызвано ограничениями со стороны вычислительных ресурсов. Поэтому, если имеется возможность сохранять большое количество решений, то зачастую не составляет труда найти множество, хорошо аппроксимирующее фронт Парето. Если же мы можем использовать только ограниченное количество решений, то полученная аппроксимация может быть недостаточно эффективной. Причем возникает дополнительный вопрос: как отличить более репрезентативное множество финальных решений от менее репрезентативных?

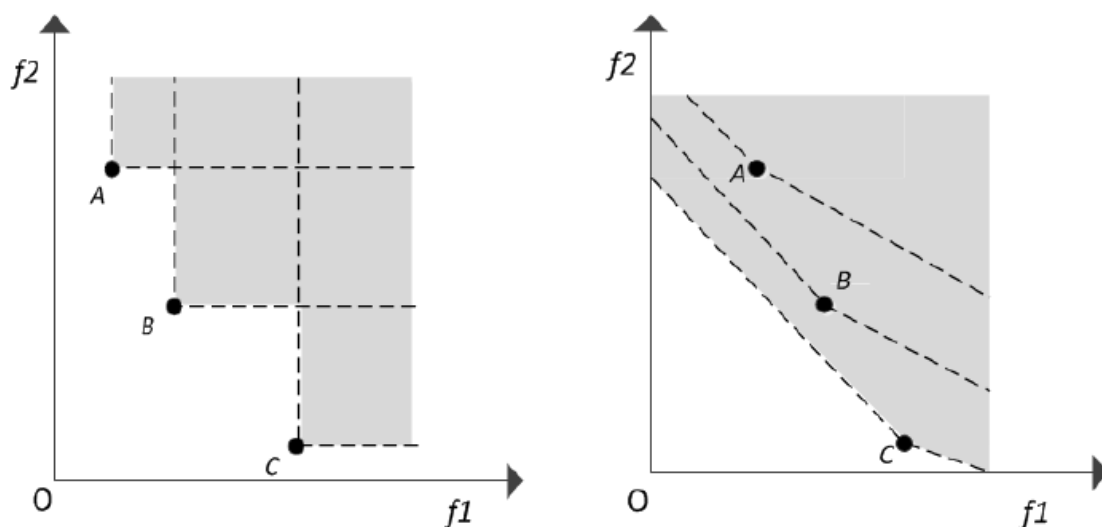


Рисунок 1.4 – Иллюстрация стандартного и расширенного отношения доминирования [129]

Sato, Aguirre и Tanaka, в 2007 г. предложили модификацию отношения Парето-доминирования [109, 110]. Область доминирования может сужаться или расширяться, согласно предпочтениям пользователя. Рисунок 1.4 показывает, что исходно решения A и B недоминируемы по Парето, но при расширении областей доминирования A становится эффективнее B (A доминирует B). Данное модифицированное отношение было включено в алгоритмическую схему метода *NSGA-II*. Результаты эксперимента показали, что *NSGA-II* с модифицированным отношением доминирования работает эффективно на задачах с большим количеством критериев [113]. Однако качество результатов всецело зависит от выбора подходящей степени расширения или сужения.

ε -доминирование – еще одно широко распространенное отношение доминирования (рисунок 1.5) [92]. Существует две схемы реализации ε -доминирования: аддитивная и мультипликативная. Говорят, что вектор $x \in R^n$ ε -доминирует вектор $y \in R^n$ (обозначается, как $x \prec_{\varepsilon} y$), тогда и только тогда (в предположении задачи минимизации), когда

аддитивная схема: $\forall i \in 1, 2, \dots, M, \quad x_i - \varepsilon \leq y_i$;

мультипликативная схема: $\forall i \in 1, 2, \dots, M, \quad x_i(1 - \varepsilon) \leq y_i$;

где M – число критериев.

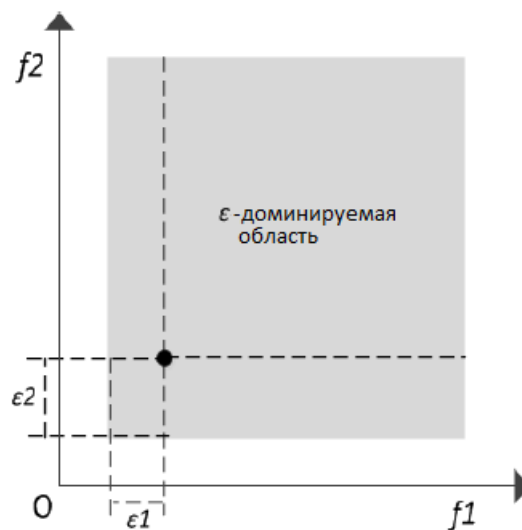


Рисунок 1.5 – Иллюстрация ε -доминирования [129]

Значение ε может быть индивидуальным для каждого критерия. Было показано, что МОЕА, основанные на ε -доминировании [48], превосходят по эффективности *NSGA-II* при решении задач с большим количеством критериев [128]. Однако, так же как и в предыдущем случае, эффективность применения методов, основанных на ε -доминировании, зависит от выбора подходящего значения ε .

Подход с использованием усредненных рангов (обозначается *AR*) был предложен исследователями Bentley и Wakefield в 1997 г. [26] и продемонстрировал эффективную работу на задачах с большим количеством целей [43]. Решения ранжируются с учетом их эффективности по каждому

критерию. В результате получается матрица ранжировок $N \times M$, где N – число индивидов популяции. Пригодность решения x описывается вектором рангов $r(x)=[r_1(x), \dots, r_M(x)]$, где $r_m(x)$ – ранг решения x по цели m . Общий ранг решения x определяется:

$$AR = \sum_{i=1}^M r_m(x) . \quad (1.2)$$

В 2001 г. Drechsler N. и Drechsler R. представил отношение предпочтения [52], согласно которому среди двух недоминируемых решений выбирается то, которое демонстрирует лучшую эффективность на большем количестве целей. Строго говоря, x_i предпочтительнее x_j (обозначается $x_i \prec_f x_j$), тогда и только тогда, когда (в предположении задачи минимизации):

$$\left| \{m : f_m(x_i) < f_m(x_j)\} \right| > \left| \{n : f_n(x_i) < f_n(x_j)\} \right| \\ \text{для } m, n \in \{1, 2, \dots, M\} . \quad (1.3)$$

Согласно результатам экспериментов, отношение предпочтения не так эффективно, как подход с использованием усредненных рангов для задач с большим количеством критериев [43].

В 2006 г. был предложен еще один метод ранжирования, основанный на подсчете побед в попарных сравнениях [94]. Для каждого решения определяется суммарное количество случаев его превосходства над другими кандидатами. Например, если решение x_i лучше решения x_j по пяти целям, но в тоже время по двум целям хуже, чем x_j , то «преимущество» x_i над x_j равно 3, а «преимущество» x_j над x_i равно -3. Результаты экспериментов показывают, что данный метод эффективен для задач со многими критериями. Однако в 2007 г. исследователями было доказано, что данный подход является эквивалентом метода AR [43].

В целом, ранжирование позволяет получить улучшенное упорядочение недоминируемых решений, и поэтому может быть применено для решения задач со многими критериями. Среди этих методов AR считается самым простым и, кроме того, одним из самых эффективных [43]. Однако Kukkonen и его коллеги

(2007 г.) показали, что методы ранжирования также не лишены недостатка, состоящего в потере разнообразия среди решений [87].

Как известно, сложности оптимизации в задачах с большим количеством целей возникают из-за высокой размерности критериального пространства. Поэтому внедрение процедур снижения размерности является естественным для МОЕА.

Deb и Saxena (2006 г.) [50] использовали линейный метод главных компонент (*англ.* Principal Component Analysis, *сокр.* PCA) для отбрасывания сильно коррелирующих критериев при оптимизации методом *NSGA-II*. Данный подход способен определить правильную комбинацию целей, которая описывает большинство особенностей проблемы. Более того, была предложена обобщенная схема, в которой и линейный, и нелинейный *PCA* применялись для сокращения количества критериев [111].

Brockhoff и Zitzler (2006 г.) разработали метод определения минимального набора критериев, основанный на сохранении структуры доминирования всех рассматриваемых решений [36].

Методы, основанные на снижении количества критериев, теряют свою эффективность при решении задач с большим числом конкурирующих целей. Результаты экспериментов показали, что данный подход не демонстрирует очевидных преимуществ по сравнению с другими алгоритмами [129].

Подходы, основанные на декомпозиции, преобразуют многокритериальную задачу к набору однокритериальных, которые получаются с помощью специальных функций (например, взвешенная сумма и функция Чебышева) с различными весовыми коэффициентами. Решения оцениваются взвешенными скалярными метриками.

Далее рассматривается ряд наиболее известных алгоритмов, основанных на декомпозиции. Заметим, что данные методы в некоторых исследованиях называются также агрегированными или методами, основанными на скалярных функциях.

Методы, основанные на декомпозиции, стали особенно популярны после выхода статьи [134], описывающей алгоритм *MOEA/D*. Причем данный подход может быть применен не только к решению задач многокритериальной, но и комбинаторной оптимизации. Было установлено, что этот метод превосходит *NSGA-II* на задачах со сложной топологией множества Парето [93]. *MOEA/D* является победителем соревнования «Эволюционные алгоритмы безусловной многокритериальной оптимизации» на конгрессе эволюционных вычислений в 2009 [135].

По сравнению с алгоритмами, реализующими принцип Парето-доминирования, методы, в основе которых лежит идея декомпозиции, обладают лучшей поисковой способностью и большей эффективностью в вычислительном смысле при оценивании пригодности [134, 135]. Однако, также известно, что качество работы алгоритмов, основанных на декомпозиции, зависит от используемых весов. Обычно определение множества весовых коэффициентов априори не является тривиальной задачей из-за нехватки знаний о структуре проблемы. В связи с чем были разработаны алгоритмы, использующие механизм адаптации весовых коэффициентов [46, 84].

Согласно представленным сведениям, в развитии эволюционных методов многокритериальной оптимизации можно выявить три основных периода [41]:

- на первом этапе предпринимались попытки приведения многокритериальной задачи к однокритериальной постановке с дальнейшим применением уже существующих эволюционных методов. Однако вследствие низкой эффективности данного подхода возникла необходимость развития алгоритмического аппарата способного оперировать сразу несколькими критериями;
- во второй половине 80-х гг. XX века начали предприниматься попытки проектирования эффективных схем многокритериальных эволюционных алгоритмов. Уже спустя 30 лет был накоплен обширный аппарат решения задач с несколькими целевыми критериями. Кроме того, были сформулированы основные принципы функционирования данных методов;

– на текущий момент эволюционные методы многокритериальной оптимизации активно используются при решении практических задач. Однако было установлено, что при увеличении числа целевых функционалов эффективность данных алгоритмов резко снижается, что стало основным вопросом, требующим решения со стороны исследователей.

ВЫВОДЫ

В данной главе представлен один из ключевых аспектов обработки речевых сигналов, а именно задача распознавания эмоций говорящего по акустическим характеристикам. Как было отмечено, стандартные методы интеллектуального анализа данных не демонстрируют желаемой эффективности на рассматриваемой задаче, поэтому в качестве альтернативных подходов в работе предлагаются технологии, основанные на эвристических процедурах.

Несмотря на ряд положительных эффектов, возникающих при интеграции эволюционных вычислений и средств машинного обучения, отмечают и недостатки, связанные с необходимостью выбора эвристики, эффективной для конкретной задачи, и большим количеством вычислительных ресурсов, требуемых для работы эволюционных методов.

Поэтому для устранения указанных недостатков был разработан коллективный параллельный генетический алгоритм для решения задач многокритериальной оптимизации, схема которого описана в следующей главе (глава 2). Далее реализованная эвристическая процедура была успешно внедрена в инструменты интеллектуального анализа данных (глава 3 и 4), применяемые для решения задачи распознавания эмоций человека по речи.

Глава 2. Разработка, реализация и исследование коллективного генетического алгоритма многокритериальной оптимизации

2.1 Основы функционирования генетических алгоритмов многокритериальной оптимизации

В общем случае принцип работы любого стохастического популяционного алгоритма базируется на трех составляющих: наличие множества кандидатов-решений, механизмы отбора наиболее эффективных точек и операторы генерирования новых решений. Эвристический поиск представляет собой итерационную процедуру, на каждой стадии которой тем или иным образом реализованы указанные составляющие.

Генетические алгоритмы (ГА) (как одно из направлений эволюционных алгоритмов) являются поисковой процедурой, оперирующей бинарными строками и основывающейся на эволюционном механизме [72]. ГА заимствуют из биологии не только идею коллективного поиска решения при помощи популяции индивидов, но и понятийный аппарат [17] (таблица 2.1).

Таблица 2.1 – Сопоставление понятийного аппарата теории эволюции и теории генетических алгоритмов

Термин из теории эволюции	Смысл понятия в теории ГА
индивид	решение-кандидат
хромосома	представление решения (для ГА это бинарная строка)
ген	часть хромосомы
пригодность	качество решения задачи
окружающая среда	проблема (задача)

ГА имитирует процесс эволюции (смены поколений) живых организмов. Общая схема работы ГА представлена следующими этапами [21] (рисунок 2.1):

1. Инициализировать начальную популяцию.
2. Оценить пригодность индивидов.

3. Проверить выполнение критерия останова (максимальное число поколений, качество найденного решения, стагнация и т.п.) Если выполняется – завершить алгоритм, в противном случае – повторять шаги 3-7.
4. Применить к текущей популяции оператор селекции, отобрав тем самым индивидов для воспроизводства. Данный механизм устроен таким образом, что отбирает тех индивидов, у которых значение пригодности выше.
5. Выполнить скрещивание (оператор рекомбинации) отобранных индивидов для получения потомков (новых решений).
6. Подвергнуть мутации генетическую информацию новых решений (с целью обновления хромосом).
7. Оценить пригодность полученных индивидов. Сформировать новое поколение.

Подробная схема ГА, описание генетических операторов и принципов их работы представлены в [6].

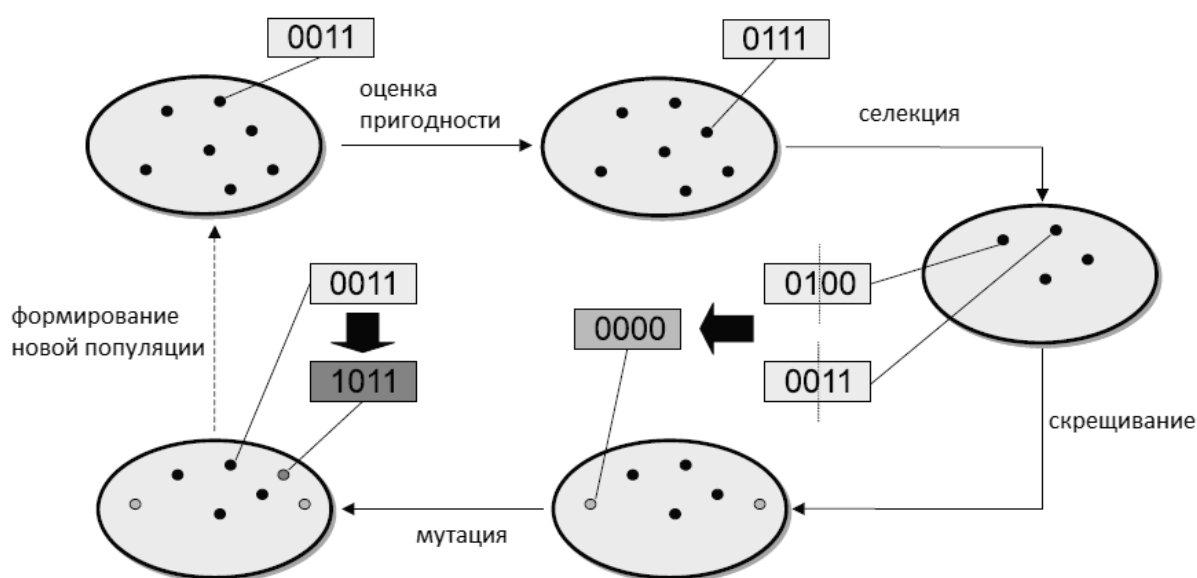


Рисунок 2.1 – Схема эволюционного алгоритма с бинарным представлением решений [138]

Как уже отмечалось, ГА оперирует популяцией кандидатов-решений, что и предопределило возможность использования данного алгоритма для отыскания репрезентативной аппроксимации множества Парето в задачах многокритериальной оптимизации. Однако применение эволюционных методов для данного класса задач сопряжено с рядом открытых вопросов.

Во-первых, необходимо выбрать эффективную процедуру назначения пригодности, поскольку данная скалярная величина учитывает значения сразу нескольких критериев качества. Один из самых распространенных способов оценки пригодности – агрегирование критериев, а именно взвешенная сумма. В ходе работы алгоритма весам присваиваются различные значения с целью поиска множества недоминируемых точек, а не единственного решения [66, 78]. Другой подход основан на выборе критерия, с учетом значений которого происходит селекция наиболее пригодных индивидов. Так, в методе VEGA [114] отбор индивидов по каждому критерию осуществляется в равных пропорциях. Кроме того, были разработаны подходы, в которых критериям ставились в соответствие вероятности их использования для оценки пригодности: вероятности могут назначаться пользователем, выбираться случайным образом или адаптироваться в ходе работы алгоритма [88]. Однако зачастую самой эффективной признается стратегия, основанная на идее Парето-доминирования. Пригодность индивида назначается исходя из числа решений, доминирующих рассматриваемую точку [56], или числа кандидатов, доминируемых данным индивидом.

Второй вопрос связан с выбором метода поддержания разнообразия решений в популяции с целью получения репрезентативной аппроксимации множества или фронта Парето. Информация о густоте решений в отдельных регионах поискового пространства также используется на этапе селективного отбора. Первый способ оценки разнообразия среди решений связан с использованием ядерных функций [118]. Сначала для каждого индивида вычисляется расстояние до всех точек популяции, а затем полученные значения передаются в ядерную функцию в качестве аргумента. Суммарная величина представляет оценку плотности по каждому индивиду. Во втором способе для

каждого решения предлагается вычислить расстояние до ближайшего k -ого соседа [118]. Обратная к полученному значению величина является оценкой плотности точек в окрестности рассматриваемого индивида. Еще одна стратегия заключается в погружении пространства в гиперсетку и вычислении количества индивидов в каждом подкубе [118]. Число соседей в ячейке гиперсетки трактуется как густота или плотность точек в рассматриваемой области пространства. Следует отметить, что вычисление расстояний или использование гиперсетки может производиться как в пространстве решений, так и в пространстве критериев (однако второй вариант считается более предпочтительным).

Другая проблема в проектировании эволюционных методов (как однокритериальной, так и многокритериальной оптимизации) связана с выбором эффективной элитарной стратегии [138]. Нередко происходит так, что в ходе стохастического поиска индивиды с высокой пригодностью замещаются менее пригодными потомками. Для предотвращения данного эффекта на стадии формирования нового поколения популяция родителей объединяется с популяцией потомков, индивиды сортируются для сохранения N наиболее эффективных кандидатов, где N – размер популяции. Описанная стратегия может быть применена для решения оптимизационных задач как с одним критерием, так и с несколькими. Еще одним способом сохранения эффективных точек-кандидатов является создание внешнего или архивного множества. При решении задач многокритериальной оптимизации данное множество используется для хранения недоминируемых индивидов. Причем на каждом поколении происходит обновление архива с целью удаления доминируемых точек и сохранения фиксированного размера данного множества (применяются различные способы отбора репрезентативных точек).

Поэтому в ходе проектирования алгоритмов перед учеными встает ряд вопросов, касающихся стратегии вычисления пригодности, сохранения многообразия решений и способов реализации элитизма. Выбор эффективной алгоритмической схемы для конкретной задачи становится нетривиальной

задачей. В связи с чем, было решено разработать коллективный эволюционный алгоритм многокритериальной оптимизации, компоненты которого представляют собой методы, основанные на различных эвристических концепциях. Основная идея кооперации заключается в объединении полезных свойств нескольких процедур для повышения надежности и эффективности многокритериального ГА.

2.2 Коллективный генетический алгоритм многокритериальной оптимизации

Островная модель ГА [130] представляет собой кооперацию нескольких параллельно работающих алгоритмов. В данном подходе вычислительные ресурсы делятся поровну между всеми компонентами модели. Пусть M – общее число индивидов, L – число алгоритмов в кооперации, тогда M/L – количество индивидов в каждой подпопуляции. Через каждые T поколений происходит обмен наилучшими кандидатами-решениями (данный процесс называется *миграцией*). Настраиваемыми параметрами являются *размер миграции*, т.е. число индивидов, подлежащих обмену, а также *интервал миграции*, т.е. число поколений между обменами. Кроме того, островные модели могут отличаться своей топологией: в данной работе используется полносвязная схема. Это означает, что каждый алгоритм обменивается лучшими решениями со всеми компонентами кооперации.

Предполагается, что использование нескольких эволюционных методов позволит достичь большего разнообразия во множестве недоминируемых точек. Определенные свойства каждого из алгоритмов островной модели могут быть полезны на различных стадиях функционирования всей кооперации. Поэтому в качестве компонент были выбраны методы, не только демонстрирующие высокую эффективность, но и основанные на различных эвристических концепциях (рисунок 2.2) (описание коллективного алгоритма, результаты тестирования представлены также в [31]).



Рисунок 2.2 – Схема коллективного многокритериального генетического алгоритма

Далее подробно рассмотрим схемы методов NSGA-II, SPEA2 и PICEA-g, включенных в кооперацию.

Non-dominated Sorting Genetic Algorithm II (NSGA-II). Усовершенствованная версия алгоритма NSGA была предложена Дебом (Deb) в 2004 г. с целью устранения ряда недостатков, присутствующих в исходной алгоритмической схеме. В методе NSGA-II [49] используется стратегия элитизма, отсутствует необходимость настройки параметра, отвечающего за поддержание разнообразия в популяции, а также применяется более эффективный механизм сортировки решений.

На первом этапе инициализируется популяция. Далее кандидаты-решения сортируются с помощью оператора *Non-dominated sort* и каждому индивиду ставится в соответствие ранг: недоминируемые точки имеют ранг 1, точки, доминируемые только решениями с рангом 1 имеют ранг 2 и т.д. Совокупность индивидов со значением ранга i составляет i -ый фронт. Также для каждого решения оценивается его близость к соседям (*crowding distance*): чем больше это расстояние, тем разнообразнее индивиды в популяции. На каждом поколении на основании ранга и близости индивида к другим точкам осуществляется селективный отбор родителей для производства потомков. Далее применяются

операторы скрещивания и мутации. Родители и потомки объединяются в одну популяцию, сортируются с помощью оператора *Non-dominated sort*, также оценивается *crowding distance* и N лучших индивидов переходят в следующее поколение, где N – размер популяции.

Рассмотрим более подробно оператор *Non-dominated sort*, механизм назначения *crowding distance* и процедуру селекции.

Non-dominated sort

- Для каждого индивида p в популяции P выполнить:
 - присвоить $S_p = \emptyset$, где S_p – множество, содержащее точки популяции, доминируемые индивидом p ;
 - инициализировать $n_p = 0$, где n_p – количество точек популяции, доминирующих индивида p ;
 - для каждой точки q из P :
 - если $p \succ q$, то скопировать q во множество S_p , т.е. $S_p = S_p \cup \{q\}$;
 - если $p \prec q$, то инкрементировать n_p , т.е. $n_p = n_p + 1$;
 - если $n_p = 0$, то нет индивидов, доминирующих p , значит, p принадлежит первому фронту; присвоить решению p ранг 1, обновить первый фронт $F_1 = F_1 \cup \{p\}$.
- Инициализировать счетчик $i = 1$.
- Выполнять, пока $F_i \neq \emptyset$:
 - Присвоить $Q = \emptyset$, где Q – множество для хранения индивидов $(i + 1)$ -ого фронта.
 - Для каждого индивида p фронта F_i :
 - для каждого индивида q из S_p :
 - $n_q = n_q - 1$;
 - если $n_q = 0$, то $q_{rank} = i + 1$.
 - Обновить множество Q : $Q = Q \cup \{q\}$.

- Инкрементировать счетчик $i = i + 1$.
- Присвоить $F_i = Q$, т.е. инициализировать следующий фронт.

Данный алгоритм сортировки отличается от оригинальной процедуры, представленной в NSGA, использованием информации об индивидах S_p , доминируемых рассматриваемым решением p , а также количеством точек, доминирующих p .

Назначение crowding distance

Для сравнения индивидов, принадлежащих фронту F_i , используется мера близости к соседним точкам.

- Для каждого фронта F_i :
 - инициализировать *crowding distance* для всех индивидов фронта:

$$F_i(d_j) = 0$$
, где j соответствует j -ому индивиду фронта F_i ;
 - для каждого критерия m :
 - сортировать индивидов фронта F_i по значениям критерия m :

$$I = \text{sort}(F_i, m);$$
 - первому и последнему индивиду в отсортированном списке поставить в соответствие бесконечное значение близости: $I(d_1) = \infty$ и $I(d_n) = \infty$, где n – число точек фронта F_i ;
 - для $k = 2 \dots (n - 1)$:
 - $$I(d_k) = I(d_k) + \frac{I(k+1).m - I(k-1).m}{f_m^{\max} - f_m^{\min}};$$
 - $I(k).m$ – значение m -ого критерия k -ого индивида в списке I .

Основная идея заключается в оценке расстояний между индивидами фронта F_i по значениям целевых критериев с использованием евклидовой метрики. Индивидам с граничными значениями критериев присваивается бесконечно большое значение *crowding distance*.

В методе NSGA-II используется бинарная турнирная селекция. Сравнение индивидов осуществляется с учетом рангов и близости к соседним точкам:

- при сравнении индивидов с разными значениями рангов (т.е. номерами фронтов) выбирается тот, чей ранг меньше;
 - при сравнении индивидов, принадлежащих одному фронту, выбирается тот, чья метрика *crowding distance* больше.
-

Таким образом, пригодность индивидов в методе NSGA-II оценивается на основании не только принципа Парето-доминирования, но и с использованием близости решений в пространстве критериев. Данный механизм назначения пригодности направлен на поддержание разнообразия кандидатов-решений в популяции. Кроме того, в методе заложена стратегия элитизма, заключающаяся в возможности перехода родительских особей с высокой пригодностью в следующее поколение.

Strength Pareto Evolutionary Algorithm2 (SPEA2). В 1999 г. Цитцлером и Тиле (Zitzler и Thiele) был предложен метод SPEA2 как усовершенствованная модификация алгоритма SPEA [139].

В данном методе недоминируемые индивиды копируются во внешнее (архивное) множество, мощность которого регулируется механизмом кластеризации. Значения функции пригодности определяются в соответствии с концепцией Парето-доминирования, и, несмотря на то, что «лучшие» индивиды хранятся отдельно во внешнем множестве, они участвуют в селекции.

Введем обозначения:

P_t и \bar{P}_t – популяция и архивное множество на t -ом поколении, N и \bar{N} – размер популяции и архива, T – максимальное число поколений.

Схема метода SPEA2 включает в себя следующие этапы:

1. Инициализировать начальную популяцию P_0 случайным образом и

создать пустое внешнее множество \bar{P}_0 . Положить $t = 0$ – номер поколения.

2. Вычислить пригодность индивидов текущей популяции и внешнего множества.

3. Скопировать всех недоминируемых индивидов из P_t и \bar{P}_t в \bar{P}_{t+1} . Если мощность полученного множества превышает установленное значение \bar{N} , то применить процедуру кластеризации. Если же мощность множества \bar{P}_{t+1} меньше \bar{N} , то необходимо доукомплектовать \bar{P}_{t+1} наиболее пригодными, доминируемыми индивидами из P_t .

4. Если $t \geq T$ (или же выполняется другой критерий останова), то алгоритм заканчивает работу. Множество $P^* = \bar{P}_{t+1}$ – искомая аппроксимация множества Парето.

5. С помощью оператора селекции из текущей популяции и внешнего множества отобрать индивидов для воспроизводства.

6. Получить новых индивидов с использованием оператора рекомбинации. Сформировать новую популяцию P_{t+1} из потомков.

7. Использовать механизм мутации для обновления генетической информации в популяции.

Таким образом, результатом работы метода SPEA2 является аппроксимация множества Парето.

Оценка пригодности индивидов популяции и архивного множества

Каждой точке из архивного множества \bar{P}_t и текущей популяции P_t ставится в соответствие величина $S(\bar{x}_i)$, называемая «силой» и равная числу решений, доминируемых рассматриваемой точкой:

$$S(\bar{x}_i) = |\{\bar{x}_j \mid \bar{x}_j \in P_t \cup \bar{P}_t \wedge \bar{x}_i \succ \bar{x}_j\}|. \quad (2.1)$$

На основе полученных значений $S(\bar{x}_i)$ оценивается «сырая» пригодность:

$$R(\bar{x}_i) = \sum_{\bar{x}_j \in P_t \cup \bar{P}_t \wedge \bar{x}_i \succ \bar{x}_j} S(\bar{x}_j). \quad (2.2)$$

В данном методе пригодность минимизируется, т.е. $R(\bar{x}_i) = 0$ соответствует недоминируемым индивидам. Чем больше значение $R(\bar{x}_i)$, тем большим количеством точек доминируется \bar{x}_i .

Несмотря на то, что назначение «сырой» пригодности реализует идею Парето-доминирования, может сложиться ситуация, когда нескольким индивидам соответствует одинаковое значение $R(\bar{x}_i)$. Чтобы избежать такой ситуации, в функцию пригодности вводится компонента, учитывающая плотность точек в пространстве критериев на основе метода k -ближайших соседей. Для каждого индивида по евклидовой метрике оцениваются расстояния до кандидатов-решений из текущей популяции и внешнего множества. Далее полученные расстояния упорядочиваются по возрастанию, и для каждого индивида определяется значение σ_i^k , т.е. расстояние до ближайшего k -ого соседа. Зачастую значение k определяется как: $k = \sqrt{N + \bar{N}}$.

После этого оценивается «плотность» $D(\bar{x}_i)$:

$$D(\bar{x}_i) = \frac{1}{\sigma_i^{k+2}}. \quad (2.3)$$

Сумма в знаменателе гарантирует, что $D(\bar{x}_i) < 1$. В конечном счете, пригодность $F(\bar{x}_i)$ индивидов складывается из «плотности» $D(\bar{x}_i)$ и «сырой» пригодности $R(\bar{x}_i)$.

$$F(\bar{x}_i) = D(\bar{x}_i) + R(\bar{x}_i). \quad (2.4)$$

В методе SPEA2 применяется турнирная селекция, размер турнира равен 2. В отборе участвуют не только индивиды текущей популяции, но и внешнего множества. Случайным образом выбираются две особи, победителем в турнире признается индивид с меньшей пригодностью.

Процедура кластеризации для сокращения архива

Для того чтобы размер внешнего множества оставался фиксированным, на шаге 3 применяется механизм кластеризации. ЛПР может задать желаемое число альтернатив, которые будут представлены в качестве решения задачи.

Процедура кластеризации применяется при условии $|\bar{P}_{t+1}| > \bar{N}$. На каждой итерации из архивного множества удаляется по одному индивиду до достижения $|\bar{P}_{t+1}| = \bar{N}$: для удаления выбирается та точка \bar{x}_i , для которой $\bar{x}_i \leq_d \bar{x}_j, \forall \bar{x}_j \in \bar{P}_{t+1}$:

$$\bar{x}_i \leq_d \bar{x}_j: \forall 0 < k < |\bar{P}_{t+1}| : \sigma_i^k = \sigma_j^k \vee \exists 0 < k < |\bar{P}_{t+1}| : [(\forall 0 < l < k : \sigma_i^l = \sigma_j^l) \wedge \sigma_i^k < \sigma_j^k]. \quad (2.5)$$

где σ_i^k – расстояние от \bar{x}_i до k -ого ближайшего соседа из \bar{P}_{t+1} в пространстве критериев. Другими словами, на каждой стадии выбирается индивид с минимальным расстоянием до одного из своих соседей.

Следовательно, в методе SPEA2 идея элитизма реализуется через хранение недоминируемых индивидов в архивном множестве. Пригодность кандидатов-решений оценивается на основе принципа Парето-доминирования и плотности этих точек в пространстве критериев. Для сокращения архива, а также поддержания разнообразия среди недоминируемых индивидов используется процедура кластеризации.

Preference-Inspired Co-Evolutionary Algorithm using goal vectors (PICEA-g). Алгоритм PICEA-g [129] был предложен Р. Вангом (R. Wang) в 2013 г. Рассмотрим схему детально схему метода.

Введем обозначения:

S – популяция кандидатов-решений мощности N , G – множество целевых векторов мощности N_g , $maxGen$ – максимальное число поколений, M – количество целевых критериев, α – скалярный параметр, $ASize$ – размер архива, $BestF$ – архивное множество, F_S – массив со значениями критериев для точек популяции, $GBounds$ – граничные значения интервалов по каждому из критериев для генерирования целевых векторов.

1. Инициализировать популяцию S случайным образом.

2. Вычислить значения критериев F_S для кандидатов-решений. Оценить пригодность индивидов S (механизм оценки пригодности представлен далее).

3. Скопировать недоминируемых индивидов в архив $BestF$. Определить границы для генерирования целевых векторов G_{Bounds} . Сгенерировать случайным образом множество целевых векторов G в границах G_{Bounds} (процедура определения интервалов генерирования целевых векторов описана ниже).

4. Для $t=1 \dots maxGex$ выполнить:

4.1. Применить к текущей популяции S генетические операторы (селекция, скрещивание, мутация) с целью получения потомков Sc .

4.2. Вычислить значения целевых критериев F_{Sc} для популяции потомков.

4.3. Объединить родителей S и потомков Sc во множество $JointS$.

4.4. Сгенерировать дополнительное множество целевых векторов Gc мощности N_g в границах G_{Bounds} . Объединить целевые векторы G и Gc в одно множество $JointG$.

4.5. Вычислить пригодности индивидов $JointS$ и векторов $JointG$.

4.6. Сформировать новую популяцию S : скопировать недоминируемых индивидов из $JointS$ в S и доукомплектовать S наилучшими доминируемыми решениями на основании пригодности.

4.7. Выбрать N_g наиболее пригодных векторов из $JointG$ для обновления G .

4.8. Обновить архив $BestF$ (скопировать вновь полученные недоминируемые решения и удалить доминируемые точки). В случае если после обновления размер архива превышает допустимое значение, применить процедуру кластеризации (аналогично методу SPEA2).

4.9. Определить новые границы G_{Bounds} для генерирования целевых векторов.

Недоминируемые точки архивного множества $BestF$ представляют искомую аппроксимацию множества Парето.

Оценка пригодности индивидов популяции и целевых векторов

В основе вычисления пригодности индивидов и целевых векторов лежит принцип Парето-доминирования. Пригодность кандидатов-решений определяется как:

$$Fit_s = \sum_{g \in G \cup G_c | s \succ g} \frac{1}{n_g}, \quad (2.6)$$

где n_g – число индивидов популяции, доминирующей целевой вектор g .

Пригодность вектора g вычисляется по формуле:

$$Fit_g = \frac{1}{1 + \gamma}, \quad (2.7)$$

$$\gamma = \begin{cases} 1, & n_g = 0, \\ \frac{n_g - 1}{2N - 1}, & \text{иначе.} \end{cases}$$

Механизм назначения пригодности рассмотрим на примере двухкритериальной задачи минимизации.

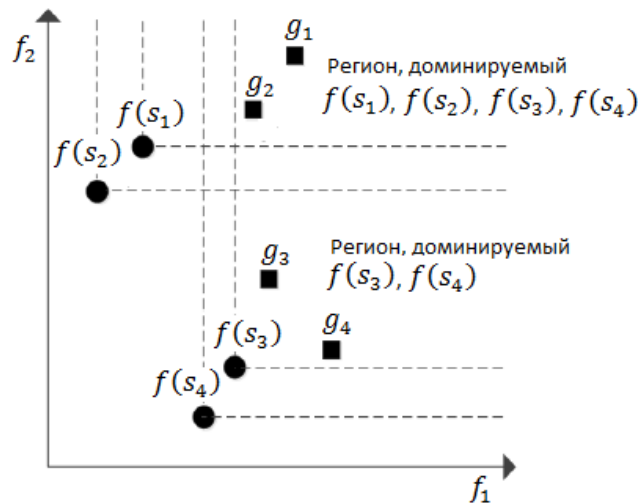


Рисунок 2.3 – Пример двухкритериального пространства поиска [129]

Пусть на текущей итерации (рисунок 2.3) имеются кандидаты-решения s_1 и s_3 , их потомки s_2 и s_4 , целевые векторы g_1 и g_3 , а также дополнительно сгенерированные векторы g_2 и g_4 , $N=N_g=2$.

В рассматриваемом примере s_3 и s_4 доминируют g_3 и g_4 , значит, $n_{g_3} = n_{g_4} = 2$. Векторы g_1 и g_2 , принадлежат региону, доминируемому индивидами s_1, s_2, s_3 и s_4 , поэтому $n_{g_1} = n_{g_2} = 4$.

Следовательно, пригодность кандидатов-решений равна:

$$Fit_{s_1} = Fit_{s_2} = \frac{1}{n_{g_1}} + \frac{1}{n_{g_2}} = \frac{1}{4} + \frac{1}{4} = \frac{1}{2},$$

$$Fit_{s_3} = Fit_{s_4} = \frac{1}{n_{g_1}} + \frac{1}{n_{g_2}} + \frac{1}{n_{g_3}} + \frac{1}{n_{g_4}} = \frac{1}{4} + \frac{1}{4} + \frac{1}{2} + \frac{1}{2} = \frac{3}{2}.$$

Пригодность целевых векторов:

$$\gamma_{g_1} = \frac{n_{g_1}-1}{2N-1} = \frac{4-1}{4-1} = 1, \text{ аналогично, } \gamma_{g_2} = 1, \gamma_{g_3} = \gamma_{g_4} = \frac{2-1}{4-1} = \frac{1}{3}.$$

$$Fit_{g_1} = \frac{1}{\gamma_{g_1}+1} = \frac{1}{2}, Fit_{g_2} = \frac{1}{2}, Fit_{g_3} = Fit_{g_4} = \frac{1}{\frac{1}{3}+1} = \frac{3}{4}.$$

Определение интервалов генерирования целевых векторов

Интервалы $GBounds$ для генерирования целевых векторов G определяются на основании архивного множества $BestF$, где нижняя и верхняя грани по каждому критерию определяются как:

$$\begin{aligned} g_{min} &= \min(BestF_i), \\ \Delta F_i &= \max(BestF_i) - \min(BestF_i), \\ g_{max} &= \min(BestF_i) + \alpha \cdot \Delta F_i, \\ \alpha &\geq 1, i = 1, 2, \dots, M. \end{aligned} \tag{2.8}$$

Рекомендуемое значение $\alpha = 1.2$. Полученные границы g_{min} и g_{max} задают в пространстве критериев интервалы $GBounds$, в пределах которых случайным образом генерируются целевые векторы G .

Таким образом, в методе PCEA-g реализуется сразу две стратегии элитизма: хранение недоминируемых индивидов в архивном множестве и возможность копирования «родителей» с высокой пригодностью в популяцию потомков. Оценка пригодности индивидов основана на принципе Парето-

доминирования с привлечением целевых векторов. Поддержание разнообразия среди недоминируемых точек архивного множества осуществляется с помощью оператора кластеризации (подобно методу SPEA2).

В таблице 2.2 приведены особенности каждого из рассмотренных эволюционных многокритериальных алгоритмов.

Таблица 2.2. Основные отличия используемых алгоритмов

Алгоритм	Назначение пригодности	Сохранение разнообразия	Элитизм
NSGA-II	Принцип основан на идее Парето-доминирования (механизм введения ниш) и оценке разнообразия решений (вычисление расстояний в пространстве критериев)	В рамках подхода оценивается густота точек (crowding distance)	Копирование лучших индивидов в следующее поколение
SPEA2	Принцип Парето-доминирования используется в совокупности с механизмом оценки плотности решений (расстояние до k-ого ближайшего соседа в пространстве критериев)	Механизм основан на определении расстояний до ближайших соседей	Сохранение недоминируемых решений в архивном множестве
PICEA-g	Принцип Парето-доминирования реализуется с помощью генерирования целевых векторов	Механизм основан на определении расстояний до ближайших соседей	Сохранение недоминируемых решений в архивном множестве; копирование лучших индивидов в следующее поколение

Реализация островной модели сводится к запуску параллельных процессов, каждый из которых ассоциирован с работой одного из методов NSGA-II, PICEA-g или SPEA2. Для синхронизации островов используется дополнительный процесс, который во время миграции сначала получает от основных процессов набор индивидов, подлежащих обмену, а затем пересылает их нужным компонентам. Данный вспомогательный процесс формирует финальное решение – аппроксимацию множества Парето.

Анализ алгоритмических схем NSGA-II, SPEA2 и PICEA-g показал, что в основу их функционирования заложены различные эвристические концепции. Значит, объединение данных методов в кооперацию позволит решить вопрос выбора эффективного алгоритма для рассматриваемой задачи.

Кроме того, можно предположить, что интеграция различных полезных свойств эволюционных методов в одну модель приводит к повышению надежности эвристической оптимизационной процедуры.

2.3 Тестовые задачи для исследования эволюционных алгоритмов многокритериальной оптимизации

Для исследования эффективности разработанного подхода использовались тестовые задачи с оптимизируемыми функциями тридцати переменных (семь задач с двумя критериями, три – с тремя). Эти задачи были разработаны в научной среде для сравнения эффективности реализованных алгоритмов многокритериальной оптимизации [136]:

1. Тестовая задача №1:

$$\begin{cases} f_1 = x_1 + \frac{2}{|J_1|} \sum_{j \in J_1} (x_j - \sin(6\pi x_1 + \frac{j\pi}{n}))^2 \rightarrow \min, \\ f_2 = 1 - \sqrt{x_1} + \frac{2}{|J_2|} \sum_{j \in J_2} (x_j - \sin(6\pi x_1 + \frac{j\pi}{n}))^2 \rightarrow \min, \end{cases} \quad (2.9)$$

где $n = 30$, $J_1 = \{j | j - \text{нечетное}, 2 \leq j \leq n\}$ и $J_2 = \{j | j - \text{четное}, 2 \leq j \leq n\}$.

Область определения: $\bar{x} \in [0,1] \cdot [-1,1]^{n-1}$.

Множество Парето: $0 \leq x_1 \leq 1$, $x_j = \sin(6\pi x_1 + \frac{j\pi}{n})$, $j = 2, \dots, n$.

Фронт Парето: $f_2 = 1 - \sqrt{f_1}$, $0 \leq f_1 \leq 1$.

2. Тестовая задача №2:

$$\begin{cases} f_1 = x_1 + \frac{2}{|J_1|} \sum_{j \in J_1} y_j^2 \rightarrow \min, \\ f_2 = 1 - \sqrt{x_1} + \frac{2}{|J_2|} \sum_{j \in J_2} y_j^2 \rightarrow \min, \end{cases} \quad (2.10)$$

где $n = 30$, $J_1 = \{j | j - \text{нечетное}, 2 \leq j \leq n\}$ и $J_2 = \{j | j - \text{четное}, 2 \leq j \leq n\}$.

$$y_j = \begin{cases} x_j - \left(0.3x_1^2 \cdot \cos\left(24\pi x_1 + \frac{4j\pi}{n}\right) + 0.6x_1\right) \cdot \cos\left(6\pi x_1 + \frac{j\pi}{n}\right), j \in J_1, \\ x_j - \left(0.3x_1^2 \cdot \cos\left(24\pi x_1 + \frac{4j\pi}{n}\right) + 0.6x_1\right) \cdot \sin\left(6\pi x_1 + \frac{j\pi}{n}\right), j \in J_2. \end{cases} \quad (2.11)$$

Область определения: $\bar{x} \in [0,1] \cdot [-1,1]^{n-1}$.

Множество Парето: $0 \leq x_1 \leq 1$,

$$x_j = \begin{cases} \left(0.3x_1^2 \cdot \cos\left(24\pi x_1 + \frac{4j\pi}{n}\right) + 0.6x_1\right) \cdot \cos\left(6\pi x_1 + \frac{j\pi}{n}\right), j \in J_1, \\ \left(0.3x_1^2 \cdot \cos\left(24\pi x_1 + \frac{4j\pi}{n}\right) + 0.6x_1\right) \cdot \sin\left(6\pi x_1 + \frac{j\pi}{n}\right), j \in J_2. \end{cases} \quad (2.12)$$

Фронт Парето: $f_2 = 1 - \sqrt{f_1}$, $0 \leq f_1 \leq 1$.

3. Тестовая задача №3:

$$\begin{cases} f_1 = x_1 + \frac{2}{|J_1|} \left(4 \cdot \sum_{j \in J_1} y_j^2 - 2 \cdot \prod_{j \in J_1} \cos\left(\frac{20y_j\pi}{\sqrt{j}}\right) + 2\right) \rightarrow \min, \\ f_2 = 1 - \sqrt{x_1} + \frac{2}{|J_2|} \left(4 \cdot \sum_{j \in J_2} y_j^2 - 2 \cdot \prod_{j \in J_2} \cos\left(\frac{20y_j\pi}{\sqrt{j}}\right) + 2\right) \rightarrow \min, \end{cases} \quad (2.13)$$

где $n = 30$, $J_1 = \{j | j - \text{нечетное}, 2 \leq j \leq n\}$ и $J_2 = \{j | j - \text{четное}, 2 \leq j \leq n\}$,

$$y_j = x_j - x_1^{0.5(1 + \frac{3(j-2)}{n-2})}, j = 2, \dots, n.$$

Область определения: $\bar{x} \in [0,1]^n$.

Множество Парето: $0 \leq x_1 \leq 1$, $x_j = x_1^{0.5(1 + \frac{3(j-2)}{n-2})}$, $j = 2, \dots, n$.

Фронт Парето: $f_2 = 1 - \sqrt{f_1}$, $0 \leq f_1 \leq 1$.

4. Тестовая задача №4:

$$\begin{cases} f_1 = x_1 + \frac{2}{|J_1|} \sum_{j \in J_1} h(y_j) \rightarrow \min, \\ f_2 = 1 - x_1^2 + \frac{2}{|J_2|} \sum_{j \in J_2} h(y_j) \rightarrow \min, \end{cases} \quad (2.14)$$

где $n = 30$, $J_1 = \{j | j - \text{нечетное}, 2 \leq j \leq n\}$ и $J_2 = \{j | j - \text{четное}, 2 \leq j \leq n\}$,

$$y_j = x_j - \sin\left(6\pi x_1 + \frac{j\pi}{n}\right), j = 2, \dots, n, \quad h(t) = \frac{|t|}{1 + e^{2|t|}}.$$

Область определения: $\bar{x} \in [0,1] \cdot [-2,2]^{n-1}$.

Множество Парето: $0 \leq x_1 \leq 1$, $x_j = \sin\left(6\pi x_1 + \frac{j\pi}{n}\right)$, $j = 2, \dots, n$.

Фронт Парето: $f_2 = 1 - f_1^2$, $0 \leq f_1 \leq 1$.

5. Тестовая задача №5:

$$\begin{cases} f_1 = x_1 + \left(\frac{1}{2N} + \varepsilon\right) \cdot |\sin(2N\pi x_1)| + \frac{2}{|J_1|} \sum_{j \in J_1} h(y_j) \rightarrow \min, \\ f_2 = 1 - x_1 + \left(\frac{1}{2N} + \varepsilon\right) \cdot |\sin(2N\pi x_1)| + \frac{2}{|J_2|} \sum_{j \in J_2} h(y_j) \rightarrow \min, \end{cases} \quad (2.15)$$

где $n = 30$, $J_1 = \{j | j - \text{нечетное}, 2 \leq j \leq n\}$ и $J_2 = \{j | j - \text{четное}, 2 \leq j \leq n\}$,

$$y_j = x_j - \sin\left(6\pi x_1 + \frac{j\pi}{n}\right), j = 2, \dots, n, \quad h(t) = 2t^2 - \cos(4\pi t) + 1, \quad \varepsilon = 0.1, \quad N = 10.$$

Область определения: $\bar{x} \in [0,1] \cdot [-1,1]^{n-1}$.

Фронт Парето: $\left(\frac{i}{2N}, 1 - \frac{i}{2N}\right), i = 0, 1, \dots, 2N$.

6. Тестовая задача №6:

$$\begin{cases} f_1 = x_1 + \max\{0, 2\left(\frac{1}{2N} + \varepsilon\right) \sin(2N\pi x_1)\} + \\ + \frac{2}{|J_1|} \left(4 \cdot \sum_{j \in J_1} y_j^2 - 2 \cdot \prod_{j \in J_1} \cos\left(\frac{20y_j\pi}{\sqrt{j}}\right) + 2\right) \rightarrow \min, \\ f_2 = 1 - x_1 + \max\{0, 2\left(\frac{1}{2N} + \varepsilon\right) \sin(2N\pi x_1)\} + \\ + \frac{2}{|J_2|} \left(4 \cdot \sum_{j \in J_2} y_j^2 - 2 \cdot \prod_{j \in J_2} \cos\left(\frac{20y_j\pi}{\sqrt{j}}\right) + 2\right) \rightarrow \min, \end{cases} \quad (2.16)$$

где $n = 30$, $J_1 = \{j | j - \text{нечетное}, 2 \leq j \leq n\}$ и $J_2 = \{j | j - \text{четное}, 2 \leq j \leq n\}$,

$$y_j = x_j - \sin\left(6\pi x_1 + \frac{j\pi}{n}\right), j = 2, \dots, n, \quad \varepsilon = 0.1, \quad N = 10.$$

Область определения: $\bar{x} \in [0,1] \cdot [-1,1]^{n-1}$.

Фронт Парето: $(0,1)$ – изолированная точка и N разрывных частей:

$$f_2 = 1 - f_1, \quad f_1 \cup_{i=1}^N \left(\frac{2i-1}{2N}, \frac{2i}{2N}\right).$$

7. Тестовая задача №7:

$$\begin{cases} f_1 = \sqrt[5]{x_1} + \frac{2}{|J_1|} \sum_{j \in J_1} y_j^2 \rightarrow \min, \\ f_2 = 1 - \sqrt[5]{x_1} + \frac{2}{|J_2|} \sum_{j \in J_2} y_j^2 \rightarrow \min, \end{cases} \quad (2.17)$$

где $n = 30$, $J_1 = \{j | j - \text{нечетное}, 2 \leq j \leq n\}$ и $J_2 = \{j | j - \text{четное}, 2 \leq j \leq n\}$,

$$y_j = x_j - \sin\left(6\pi x_1 + \frac{j\pi}{n}\right), j = 2, \dots, n.$$

Область определения: $\bar{x} \in [0,1] \cdot [-1,1]^{n-1}$.

Множество Парето: $0 \leq x_1 \leq 1, x_j = \sin\left(6\pi x_1 + \frac{j\pi}{n}\right), j = 2, \dots, n$.

Фронт Парето: $f_2 = 1 - f_1, 0 \leq f_1 \leq 1$.

8. Тестовая задача №8:

$$\left\{ \begin{array}{l} f_1 = \cos(0.5x_1\pi) \cos(0.5x_2\pi) + \\ + \frac{2}{|J_1|} \sum_{j \in J_1} (x_j - 2x_2 \cdot \sin(2\pi x_1 + \frac{j\pi}{n}))^2 \rightarrow \min, \\ f_2 = \cos(0.5x_1\pi) \sin(0.5x_2\pi) + \\ + \frac{2}{|J_2|} \sum_{j \in J_2} (x_j - 2x_2 \cdot \sin(2\pi x_1 + \frac{j\pi}{n}))^2 \rightarrow \min, \\ f_3 = \sin(0.5x_1\pi) + \frac{2}{|J_3|} \sum_{j \in J_3} (x_j - 2x_2 \cdot \sin(2\pi x_1 + \frac{j\pi}{n}))^2 \rightarrow \min, \end{array} \right. \quad (2.18)$$

где $n = 30$, $J_1 = \{j | (j - 1) - \text{кратно } 3, 3 \leq j \leq n\}$, $J_2 = \{j | (j - 2) - \text{кратно } 3, 3 \leq j \leq n\}$, $J_3 = \{j | j - \text{кратно } 3, 3 \leq j \leq n\}$.

Область определения: $\bar{x} \in [0,1]^2 \cdot [-2,2]^{n-2}$.

Множество Парето: $0 \leq x_1, x_2 \leq 1, x_j = 2x_2 \sin(2\pi x_1 + \frac{j\pi}{n}), j = 3, \dots, n$.

Фронт Парето: $f_1^2 + f_2^2 + f_3^3 = 1, 0 \leq f_1, f_2, f_3 \leq 1$.

9. Тестовая задача №9:

$$\left\{ \begin{array}{l} f_1 = 0.5(\max\{0, (1 + \varepsilon)(1 - 4(2x_1 - 1)^2)\} + 2x_1)x_2 + \\ + \frac{2}{|J_1|} \sum_{j \in J_1} (x_j - 2x_2 \sin(2\pi x_1 + \frac{j\pi}{n}))^2 \rightarrow \min, \\ f_2 = 0.5(\max\{0, (1 + \varepsilon)(1 - 4(2x_1 - 1)^2)\} + 2x_1)x_2 + \\ + \frac{2}{|J_2|} \sum_{j \in J_2} (x_j - 2x_2 \sin(2\pi x_1 + \frac{j\pi}{n}))^2 \rightarrow \min, \\ f_3 = 1 - x_2 + \frac{2}{|J_3|} \sum_{j \in J_3} (x_j - 2x_2 \sin(2\pi x_1 + \frac{j\pi}{n}))^2 \rightarrow \min, \end{array} \right. \quad (2.19)$$

где $n = 30$, $J_1 = \{j | (j - 1) - \text{кратно } 3, 3 \leq j \leq n\}$, $J_2 = \{j | (j - 2) - \text{кратно } 3, 3 \leq j \leq n\}$, $J_3 = \{j | j - \text{кратно } 3, 3 \leq j \leq n\}$, $\varepsilon = 0.1$.

Область определения: $\bar{x} \in [0,1]^2 \cdot [-2,2]^{n-2}$.

Множество Парето: $x_1 \in [0,0.25] \cup [0.75, 1], 0 \leq x_2 \leq 1, x_j = 2x_2 \sin(2\pi x_1 + \frac{j\pi}{n}), j = 3, \dots, n$.

Фронт Парето: $0 \leq f_3 \leq 1, 0 \leq f_1 \leq \frac{1}{4}(1 - f_3), f_2 = 1 - f_1 - f_3$ и $0 \leq f_3 \leq 1, \frac{3}{4}(1 - f_3) \leq f_1 \leq 1, f_2 = 1 - f_1 - f_3$.

10. Тестовая задача №10:

$$\left\{ \begin{array}{l} f_1 = \cos(0.5x_1\pi) \cos(0.5x_2\pi) + \\ + \frac{2}{|J_1|} \sum_{j \in J_1} (4y_j^2 - \cos(8\pi y_j) + 1) \rightarrow \min, \\ f_2 = \cos(0.5x_1\pi) \sin(0.5x_2\pi) + \\ + \frac{2}{|J_2|} \sum_{j \in J_2} (4y_j^2 - \cos(8\pi y_j) + 1) \rightarrow \min, \\ f_3 = \sin(0.5x_1\pi) + \frac{2}{|J_3|} \sum_{j \in J_3} (4y_j^2 - \\ - \cos(8\pi y_j) + 1) \rightarrow \min, \end{array} \right. \quad (2.20)$$

где $J_1 = \{j | (j - 1) - \text{кратно } 3, 3 \leq j \leq n\}$, $J_2 = \{j | (j - 2) - \text{кратно } 3, 3 \leq j \leq n\}$, $J_3 = \{j | j - \text{кратно } 3, 3 \leq j \leq n\}$, $y_j = x_j - 2x_2 \sin(2\pi x_1 + \frac{j\pi}{n})$, $j = 3, \dots, n$, $n = 30$.

Область определения: $\bar{x} \in [0,1]^2 \cdot [-2,2]^{n-2}$.

Множество Парето: $0 \leq x_1, x_2 \leq 1$, $x_j = 2x_2 \sin(2\pi x_1 + \frac{j\pi}{n})$, $j = 3, \dots, n$.

Фронт Парето: $f_1^2 + f_2^2 + f_3^2 = 1$, $0 \leq f_1, f_2, f_3 \leq 1$.

Результат работы алгоритма оценивался метрикой IGD (чем меньше ее значение, тем эффективность метода выше):

$$IGD(A, P^*) = \frac{\sum_{v \in P^*} d(v, A)}{|P^*|}, \quad (2.21)$$

где P^* – фронт Парето, A – аппроксимация фронта Парето (результат работы алгоритма), v – точка фронта Парето, $d(v, A)$ – минимальное расстояние между точкой $v \in P^*$ и множеством A , вычисленное по евклидовой метрике. Полученные значения метрики IGD усреднялись по 25 прогонам. При выделенном количестве ресурсов (число вычислений целевой вектор-функции не более 300 тыс.) доля просматриваемого поискового пространства не превышала $\sim 10^{-85}$.

2.4 Исследование эффективности разработанного коллективного генетического алгоритма многокритериальной оптимизации

Во всех экспериментах были установлены следующие настройки генетических алгоритмов: бинарная турнирная селекция, равномерное скрещивание и вероятность мутации $p_m=1/n$, где n – длина хромосомы.

Эксперимент 1. Для начала исследовалась эффективность каждого из представленных алгоритмов NSGA-II, PICEA-g и SPEA2 (не в кооперации). Число поколений было равно 500, число индивидов в популяции – 600. Анализ полученных результатов показал (таблица 2.3), что среди данных методов не существует алгоритма, демонстрирующего лучший результат на всех рассматриваемых задачах, причем в ряде случаев различия в метриках IGD существенны (выявлено с помощью t-критерия Стьюдента при уровне значимости $p=0.05$). Из этого следует, что неудачный выбор алгоритма может привести к снижению качества получаемых решений.

Таблица 2.3. Результаты тестирования алгоритмов. Эксперимент 1

Номер задачи	Значения метрики IGD			Результат применения t-критерия Стьюдента
	NSGA-II	PICEA-g	SPEA2	
1	0.097	0.107	0.099	Результаты NSGA-II и SPEA2 статистически не различимы
2	0.061	0.060	0.078	Результаты NSGA-II и PICEA-g статистически не различимы
3	0.191	0.222	0.326	Результаты NSGA-II превосходят PICEA-g и SPEA2 (наблюдается статическое различие)
4	0.055	0.0570	0.083	Результаты NSGA-II и PICEA-g статистически не различимы
5	0.426	0.498	0.518	Результаты NSGA-II превосходят PICEA-g и SPEA2 (наблюдается статическое различие)

Продолжение Таблицы 2.3

Номер задачи	Значения метрики IGD			Результат применения t-критерия Стьюдента
	NSGA-II	PICEA-g	SPEA2	
6	0.335	0.346	0.319	Результаты статистически не различимы
7	0.085	0.091	0.125	Результаты статистически не различимы
8	0.269	0.191	0.259	Результаты PICEA-g превосходят NSGA-II и SPEA2 (наблюдается статическое различие)
9	0.319	0.290	0.407	Результаты NSGA-II и PICEA-g статистически не различимы
10	0.626	0.421	0.534	Результаты PICEA-g превосходят NSGA-II и SPEA2 (наблюдается статическое различие)

Эксперимент 2. В рамках данного эксперимента исследовалась эффективность островной модели с тремя узлами как инструмента распараллеливания многокритериальных эволюционных алгоритмов. Во-первых, такая архитектура позволяет сократить время, затрачиваемое на выполнение программы, а, во-вторых, обмен информацией между параллельно работающими вычислительными узлами порождает дополнительный синергетический эффект, что в случае с многоэлементным алгоритмическим ядром системы означает повышение его эффективности.

При выделенном количестве ресурсов (300 тыс. вычислений целевой вектор-функции) размер миграции положили равным 50 (каждый «остров» получал 100 индивидов от двух других), интервал миграции – 25 поколений. В таблицах 2.4 – 2.6 представлены результаты проведенных экспериментов.

Таблица 2.4 – Результаты тестирования метода NSGA-II и его «островной» модификации

Номер задачи	NSGA-II		«Островной» NSGA-II	
	IGD	Время, с.	IGD	Время, с.
1	0.097	196.1	0.071	33.8
2	0.061	181.5	0.057	33.6
3	0.191	181.2	0.206	33.3
4	0.055	182.2	0.058	35.3
5	0.426	181.5	0.411	33.9
6	0.335	183.1	0.270	34.0
7	0.085	181	0.054	33.6
8	0.269	190.3	0.288	84.6
9	0.319	191.1	0.366	88.7
10	0.626	186.3	0.681	83.5

Таблица 2.5 – Результаты тестирования метода PICEA-g и его «островной» модификации

Номер задачи	PICEA-g		«Островной» PICEA-g	
	IGD	Время, с.	IGD	Время, с.
1	0.107	42.3	0.078	22.8
2	0.060	84.5	0.056	28.0
3	0.223	36.8	0.216	19.7
4	0.057	75.8	0.058	28.1
5	0.498	33.8	0.413	17.3
6	0.346	35.0	0.285	18.2
7	0.091	75.6	0.046	26.8
8	0.191	166.1	0.187	43.9
9	0.290	107.2	0.277	33.0
10	0.421	118.7	0.525	33.0

Таблица 2.6 – Результаты тестирования метода SPEA2 и его «островной» модификации

Номер задачи	SPEA2		«Островной» SPEA2	
	IGD	Время, с.	IGD	Время, с.
1	0.099	236.7	0.056	61.1
2	0.078	262.1	0.059	69.4
3	0.326	237.6	0.285	59.4
4	0.083	243.2	0.066	62.7
5	0.518	240.2	0.287	63.1
6	0.319	237.9	0.287	62.8
7	0.125	245.9	0.060	68.3
8	0.259	253.8	0.281	93.8
9	0.407	407.0	0.384	72.0
10	0.534	290.9	0.597	80.9

Анализ полученных результатов показал, что реализация многокритериальных ГА на основе островной модели эффективна, прежде всего, в смысле сокращения времени, затрачиваемого на работу алгоритма.

Кроме того, в 8 случаях из 10 для методов SPEA2 и PICEA-g наблюдается повышение эффективности. Результаты экспериментов, в которых островная модификация демонстрирует снижение эффективности, были исследованы с помощью t-критерия Стьюдента (уровень значимости $p=0.01$). Было выявлено, что только для метода NSGA-II на задачах 8 и 9 наблюдается статистическое различие значений метрик IGD стандартной версии алгоритма и его модификации. Следовательно, использование островной модели эффективно как в смысле сокращения временных затрат на работу алгоритма, так и в смысле повышения его эффективности.

Однако параллельные версии методов NSGA-II, PICEA-g и SPEA2 также демонстрируют различную эффективность на разных тестовых задачах. Поэтому

в качестве компонент островной модели было решено использовать различные многокритериальные ГА, а именно NSGA-II, PICEA-g и SPEA2.

Эксперимент 3. Аналогично предыдущему эксперименту размер миграции был равен 50, интервал миграции – 25 поколений. Общее количество поколений – 500, размер каждой популяции $600/3=200$ индивидов. При прежних настройках генетических операторов результаты усреднялись по 25 прогонам.

Таблица 2.7. Результаты тестирования коллективного алгоритма

Номер задачи	Коллективный алгоритм		Результат применения t-критерия
	IGD	Время, с.	
1	0.068	56.566	Превосходит лучшую компоненту
2	0.056	64.837	На уровне лучшей компоненты
3	0.202	55.952	На уровне лучшей компоненты
4	0.058	60.271	На уровне лучшей компоненты
5	0.338	56.391	На уровне лучшей компоненты
6	0.254	56.008	Превосходит лучшую компоненту
7	0.084	60.269	На уровне лучшей компоненты
8	0.259	87.240	На уровне второй (по эффективности) компоненты
9	0.314	78.532	На уровне лучшей компоненты
10	0.533	75.119	На уровне лучшей компоненты

В ходе исследования эффективности коллективного алгоритма на множестве задач CEC2009 с помощью статистической проверки (при уровне значимости $p=0.01$) было выявлено (таблица 2.7): на девяти задачах разработанный алгоритм не уступает по эффективности компоненте, демонстрирующей лучший результат, и даже превосходит ее в двух случаях (по

сравнению с результатами в таблице 2.3). Только на одной задаче с тремя критериями кооперация методов проигрывает лучшей компоненте и соответствует второму по эффективности алгоритму. Это значит, что предложенный подход является обоснованной альтернативой произвольному выбору алгоритма.

Кроме того, параллельная реализация приводит к сокращению времени работы алгоритма: в среднем на функционирование стандартных методов NSGA-II, SPEA2, PICEA-g затрачивалось 176 секунды (усреднение по всем задачам и трем алгоритмам), в то время как на работу коллективного алгоритма в среднем застрачивается 65 секунд. Конечно, результаты экспериментов во многом зависят от характеристик ЭВМ, но можно утверждать, что время вычислений существенно сокращается (в представленном случае в 2.7 раза).

Примеры полученных аппроксимаций фронта Парето для тестовых задач 4 и 7 представлены на рисунках 2.4 и 2.5 соответственно.

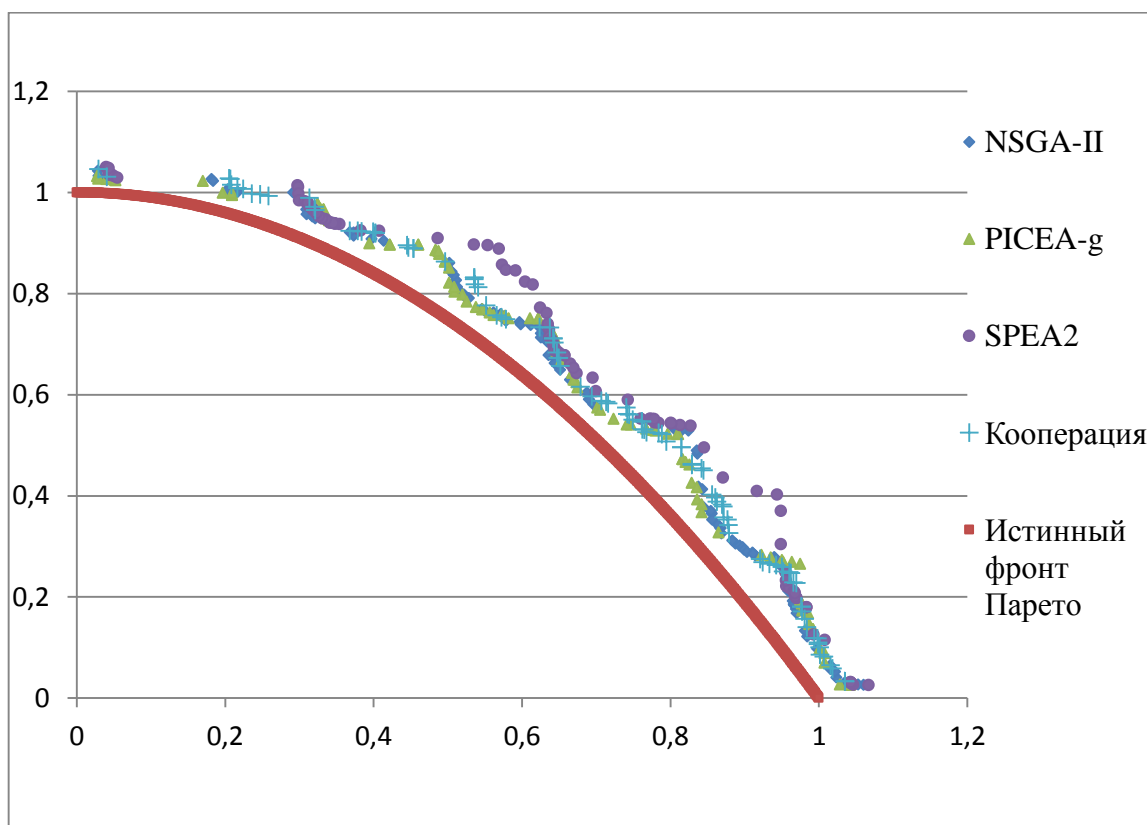


Рисунок 2.4 – Результат работы алгоритмов для Тестовой задачи 4

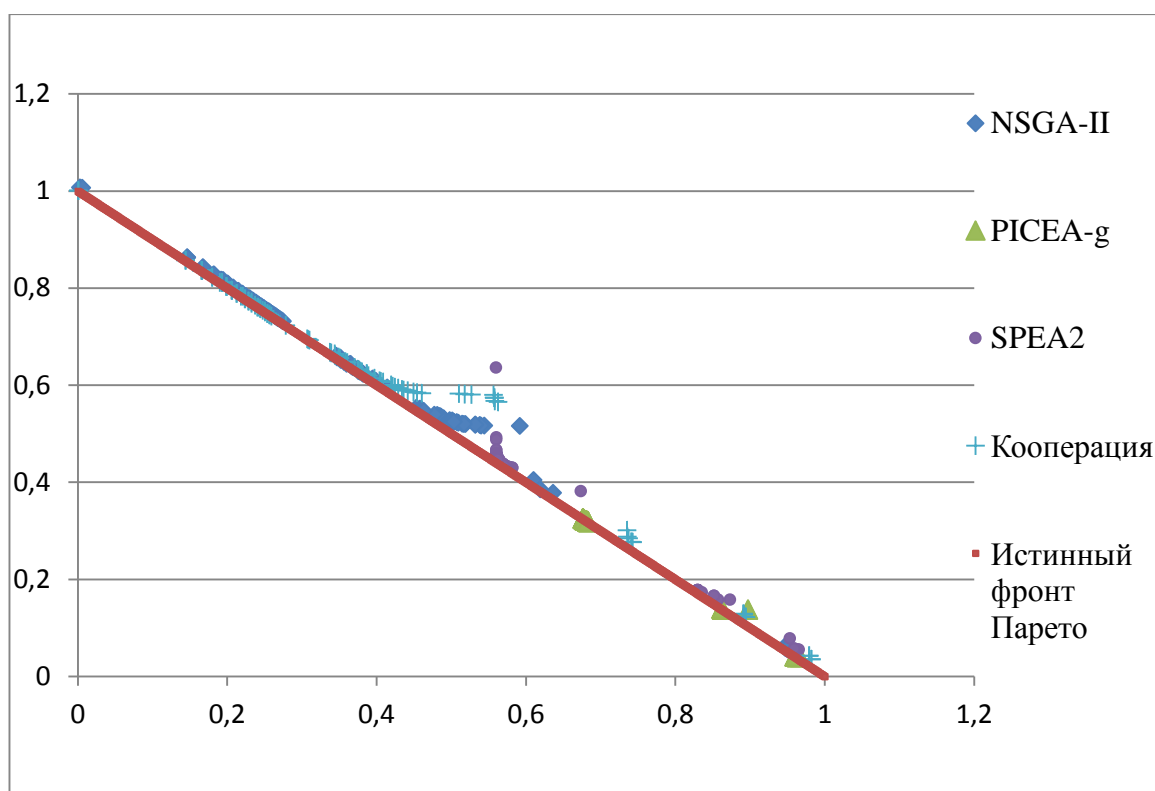


Рисунок 2.5 – Результат работы алгоритмов для Тестовой задачи 7

В Приложении 1 представлено сравнение эффективности реализованного коллективного алгоритма с победителями соревнования СЕС2009. Стоит отметить, что в разработанном методе используется стандартное бинарное кодирование, т.е. популяция представлена набором бинарных строк, что, в свою очередь, заведомо менее эффективно для решения задач вещественной оптимизации. Однако, несмотря на это, на большинстве задач алгоритм успешно конкурирует с победителями данного соревнования.

ВЫВОДЫ

С целью сокращения временных затрат и повышения надежности применяемых эволюционных технологий, был спроектирован модифицированный ГА для решения задач многокритериальной оптимизации, работа которого основана на параллельном функционировании сразу нескольких методов, реализующих различные эвристики.

В ходе статистической проверки результатов экспериментов было выявлено, что представленный коллективный алгоритм может быть успешно использован в качестве обоснованной альтернативы произвольному выбору многокритериального ГА.

Учитывая достоинства разработанного подхода, предлагается использовать его в качестве метода решения оптимизационных задач в области машинного обучения, а именно в сфере анализа речевых сигналов. В главах 3 и 4 демонстрируется успешное применение данного алгоритма к задаче распознавания эмоций человека по речи: представленная кооперация методов была использована для автоматического генерирования искусственных нейронных сетей и эвристической процедуры отбора информативных признаков.

Глава 3. Применение коллективного генетического алгоритма многокритериальной оптимизации для автоматического генерирования искусственных нейронных сетей в задаче распознавания эмоций по речи

3.1 Исследование эффективности классификаторов на задаче распознавания эмоций человека по речи

Эффективность классификаторов для задачи распознавания эмоций человека по речи была исследована на множестве баз данных (БД), включающих голосовые записи на английском, немецком и японском языках. Как было отмечено ранее, акустические характеристики извлекались с помощью программной системы OpenSMILE, разработанной специалистами Технического университета г. Мюнхена (рисунок 3.1).

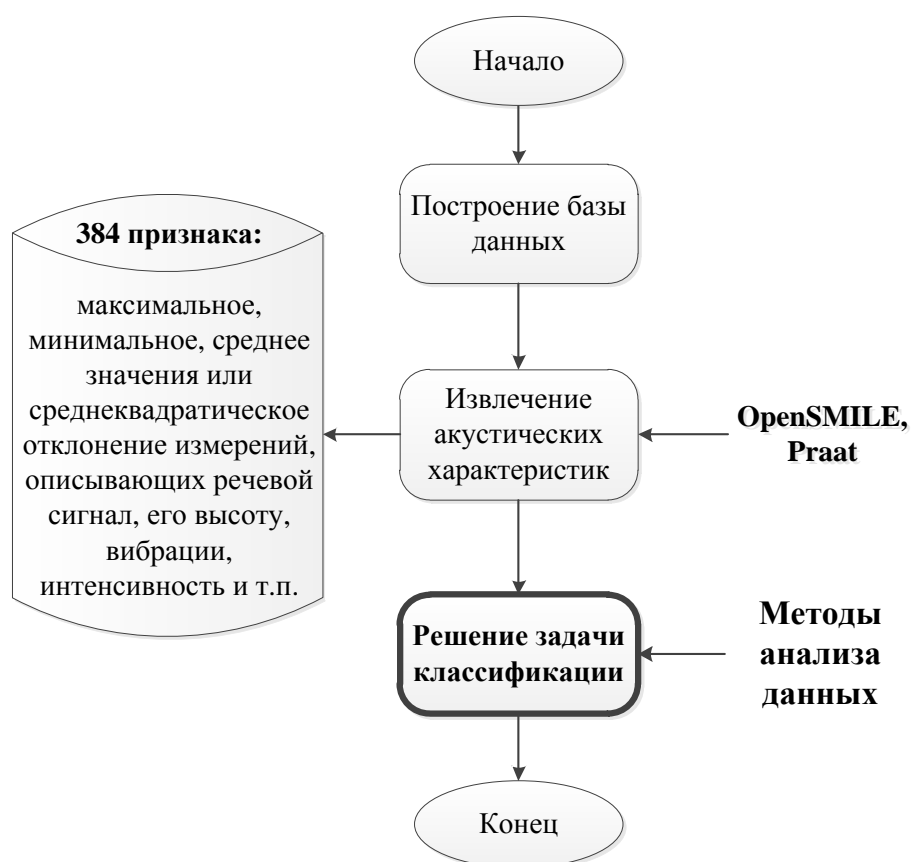


Рисунок 3.1 – Процесс распознавания эмоций человека по речи

Всего из каждой звукозаписи извлекалось 384 характеристики. В таблице 3.1 представлено статистическое описание используемых БД.

Таблица 3.1 – Описание используемых баз данных

Название БД / Характеристика	ЕМО-DB	SAVEE	LEGO	UADB
Язык	немецкий	английский	английский	японский
Суммарная продолжительность (мин.)	24.7	30.7	118.2	113.4
Количество звукозаписей	535	480	4563	4836
Средняя продолжительность звукозаписи (сек.)	2.7	3.8	1.6	1.4
Эмоции	Нейтрально, злость, страх, счастье, грусть, скука, отвращение	Злость, отвращение, страх, счастье, грусть, удивление, нейтрально	Нейтрально, злость, тишина/шум	Грусть-скука, злость-волнение, спокойствие, радость-волнение
Примечание к БД	Наигранная	Наигранная	Не наигранная	Не наигранная

База *ЕМО-DB* [38] была записана в Техническом университете Берлина и состоит из эмоциональных реплик, сказанных на немецком языке 10 актерами (5 из которых женщины). Каждое высказывание имеет один из следующих эмоциональных лейблов: нейтральный, гнев, страх, радость, печаль, скука или отвращение.

База данных *SAVEE* [70] была создана с целью исследования методов классификации эмоций на основе аудио- и видеoinформации (актерами, имитирующими эмоции, выступили четыре англоговорящих мужчины).

Звукозаписи относят к одному из следующих эмоциональных состояний: злость, отвращение, страх, счастье, грусть, удивление, нейтрально (такой набор эмоций считается стандартным).

База *LEGO* [115] включает обращения реальных пользователей в интерактивную систему голосовых ответов, используемую в автобусах Университета Карнеги-Меллон (г. Питтсбург, США). Каждый фрагмент аннотирован одним из лейблов: злость (слабая, средняя, сильная), нейтрально, дружелюбно, шум/тишина. В данной работе была произведена следующая предобработка звукозаписей: фрагменты, характеризующиеся различными уровнями злости, были объединены в один класс, а также были удалены «дружелюбные» высказывания (в связи с незначительным количеством).

База данных *UUDB* [97] состоит из спонтанной японской речи семи пар человек, в числе которых 12 женщин. В общей сложности имеется более 4 тыс. высказываний, относящихся к одному из классов: грусть-скука, злость-волнение, спокойствие, радость-волнение. Данные лейблы были получены на основании оценок трех экспертов, характеризующих каждое высказывание с позиций пяти критериев (интерес, доверие, превосходство, возбуждение и радость). В данной работе использовалась система координат лишь с двумя осями – возбуждение и радость. Начиная с первого положительного квадранта и двигаясь по часовой стрелке, области были обозначены в следующем порядке: радость-волнение, злость-волнение, грусть-скука, спокойствие.

На практике для оценки результатов работы классификатора нередко используют матрицу неточностей (*англ.* confusion matrix), столбцы которой соответствуют экспертным решениям (истинное значение класса), а строки – предсказаниям классификатора (таблица 3.2) [63]. Размерность матрицы $N \times N$, где N – число различных классов в выборке.

Таблица 3.2 – Общий вид матрицы неточностей

		Действительные значения			
		Класс ₁	Класс ₂	...	Класс _N
Предсказанные значения	Класс ₁	a ₁₁	a ₁₂	...	a _{1N}
	Класс ₂	a ₂₁	a ₂₂	...	a _{2N}

	Класс _N	a _{1N}	a _{2N}	...	a _{NN}

Матрица неточностей демонстрирует работу алгоритма и позволяет оценить его эффективность для каждого класса, содержащегося в выборке. Для этого вводятся специальные метрики *полнота* и *точность*, определяемые следующим образом. Пусть построена матрица неточностей $A=(a_{ij})$, тогда полнотой в пределах класса l (*англ. recall*) назовем величину, равную доле экземпляров данного класса, найденных классификатором, относительно всех примеров данного класса в тестовой выборке:

$$recall_l = \frac{a_{ll}}{\sum_i a_{il}} \cdot 100\%. \quad (3.1)$$

Точностью в пределах класса l (*англ. precision*) назовем величину, равную доле примеров в тестовой выборке, действительно принадлежащих классу l относительно всех экземпляров, которые были отнесены к данному классу:

$$precision_l = \frac{a_{ll}}{\sum_j a_{lj}} \cdot 100\%. \quad (3.2)$$

Чем выше точность и полнота, тем качество работы классификатора лучше. Однако при решении практических задач редко удается добиться максимальных значений обеих метрик одновременно. Поэтому часто используют такой показатель, как *F-score*, объединяющий в себе информацию и о точности, и о полноте классификатора:

$$F_score = 2 \cdot \frac{Recall \cdot Precision}{Recall + Precision}, \quad (3.3)$$

где $Recall = \sum_l recall/N$, $Precision = \sum_l precision/N$.

Пример матрицы неточностей и технология оценки метрики F_score рассмотрены далее.

В ходе эксперимента была исследована эффективность следующих классификаторов (использованы реализации данных алгоритмов в программной системе Weka 3.6.10 [67]):

- 1) **полносвязный перцептрон** (*англ.* Multilayer Perceptron, *сокр.* MLP) с одним скрытым слоем, содержащим N нейронов, функциями активации которых выступают сигмоиды. N определяется по правилу: $N = [(N_{Features} + N_{Classes})/2 + 1]$, где $N_{Features}$ – размерность входного вектора признаков, $N_{Classes}$ – количество классов в выборке. Для обучения использовался алгоритм обратного распространения ошибки;
- 2) **машины опорных векторов** (*англ.* Support Vector Machine, *сокр.* SVM), для обучения которых применялся метод последовательной минимальной оптимизации Дж. Платта [4, 82, 102];
- 3) **логистическая регрессия** (*англ.* Logistic Regression, *сокр.* Logit) оперирует апостериорными вероятностями, для моделирования которых используется линейная функция [90];
- 4) **радиально-базисная нейронная сеть** (*англ.* Radial Basis Function Network, *сокр.* RBF) представляет собой модель, активационными функциями которой выступают функция Гаусса [37];
- 5) **наивный Байесовский классификатор** (*англ.* Naive Bayes) основывается на теореме Байеса и работает в предположении, что все пары атрибутов являются независимыми [81];
- б) **деревья решений**, для построения которых использовался алгоритм J48 (алгоритм J48 – это модификация метода C4.5, реализованная в форме свободно распространяемого программного обеспечения) [105, 106];

- 7) **ансамбль деревьев решений** (*англ.* Random Forest) проектируется с использованием идеи «бэггинга» и метода случайных подпространств [28];
- 8) **бэггинг** (*англ.* Bagging) – группа мета-алгоритмов, отличительным свойством которых является попытка избежать переобучения [29];
- 9) **аддитивная логистическая регрессия** (*англ.* Additive Logistic Regression, *сокр.* LogitBoost) использует подход «бустинг» для построения модели [60];
- 10) алгоритм генерирования правил **1R** (*англ.* One Rule) предназначен для выявления одного атрибута, классификация по которому обеспечивает наименьшую ошибку [73].

Для сравнения эффективности работы классификаторов была использована процедура кросс-валидации: выборки случайным образом делились на 6 стратифицированных частей (т.е. с сохранением во всех частях первоначального процентного соотношения измерений, принадлежащих разным классам), каждая из частей по очереди использовалась в качестве тестовых примеров. Результаты эксперимента представлены на рисунках 3.2-3.5

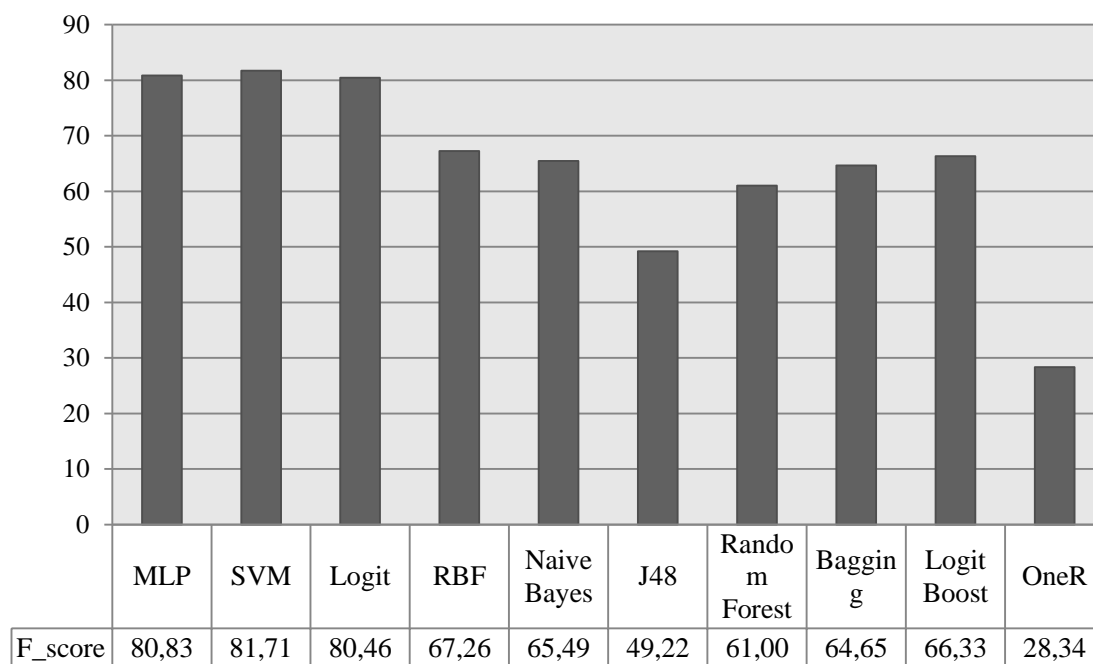


Рисунок 3.2 – Результаты работы классификаторов на БД *EMO-DB* (*F-score*, %)

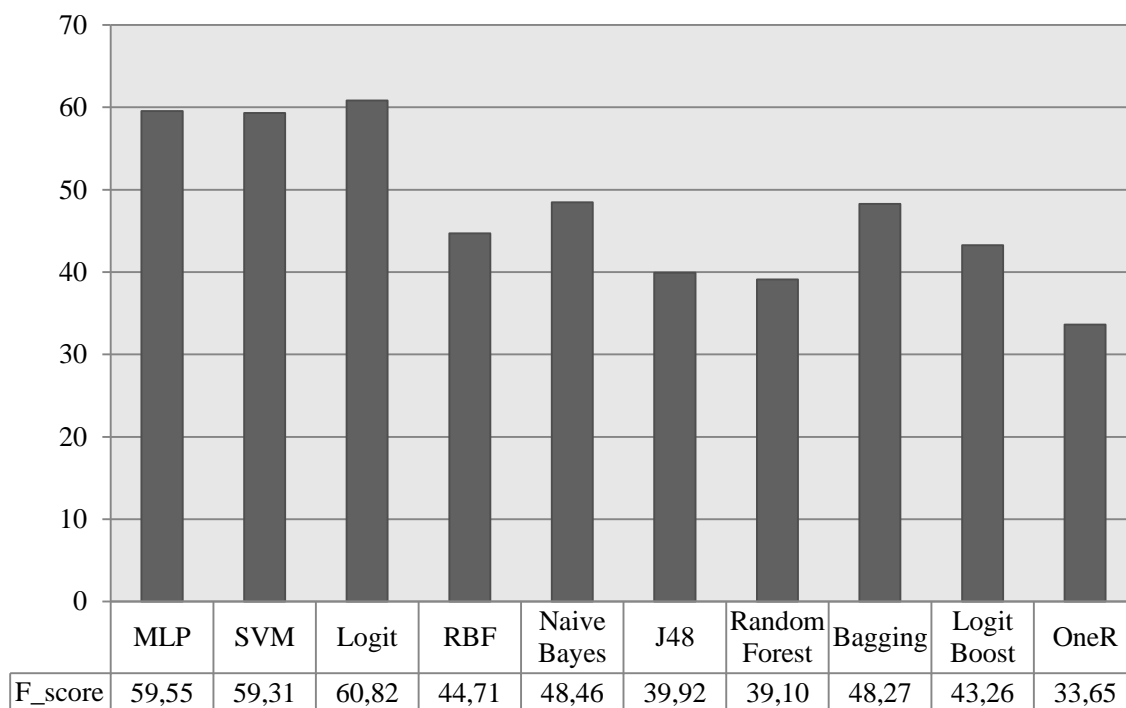


Рисунок 3.3 – Результаты работы классификаторов на БД *SAVEE* (*F-score*, %)

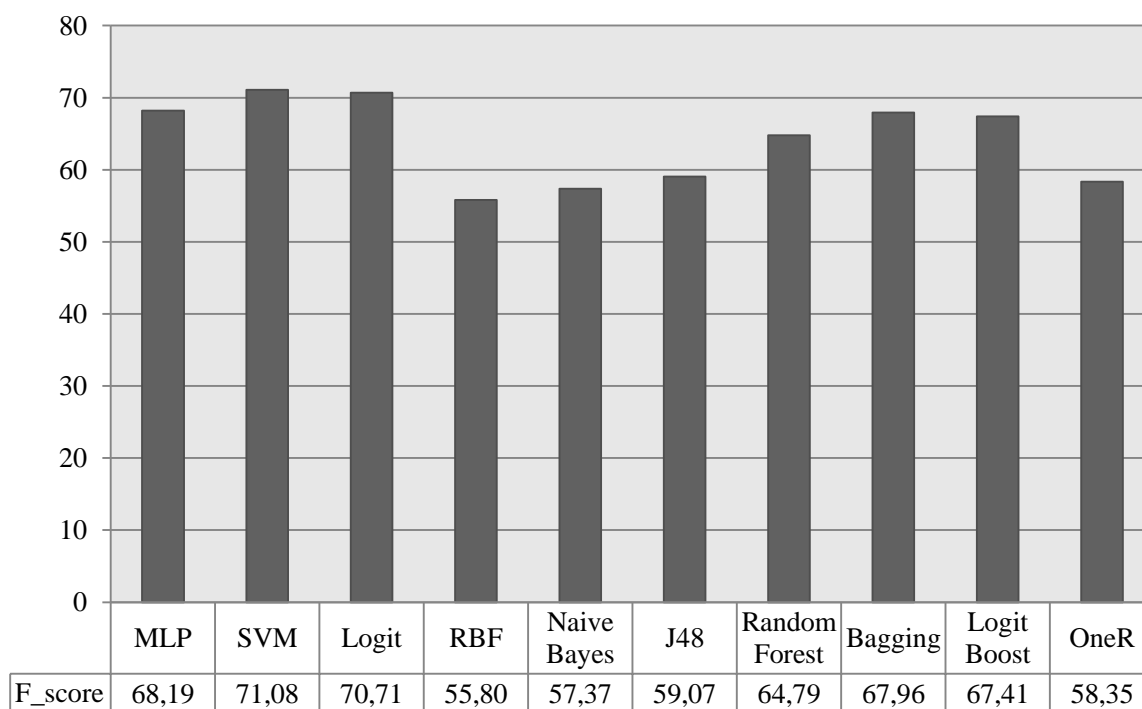


Рисунок 3.4 – Результаты работы классификаторов на БД *LEGO* (*F-score*, %)

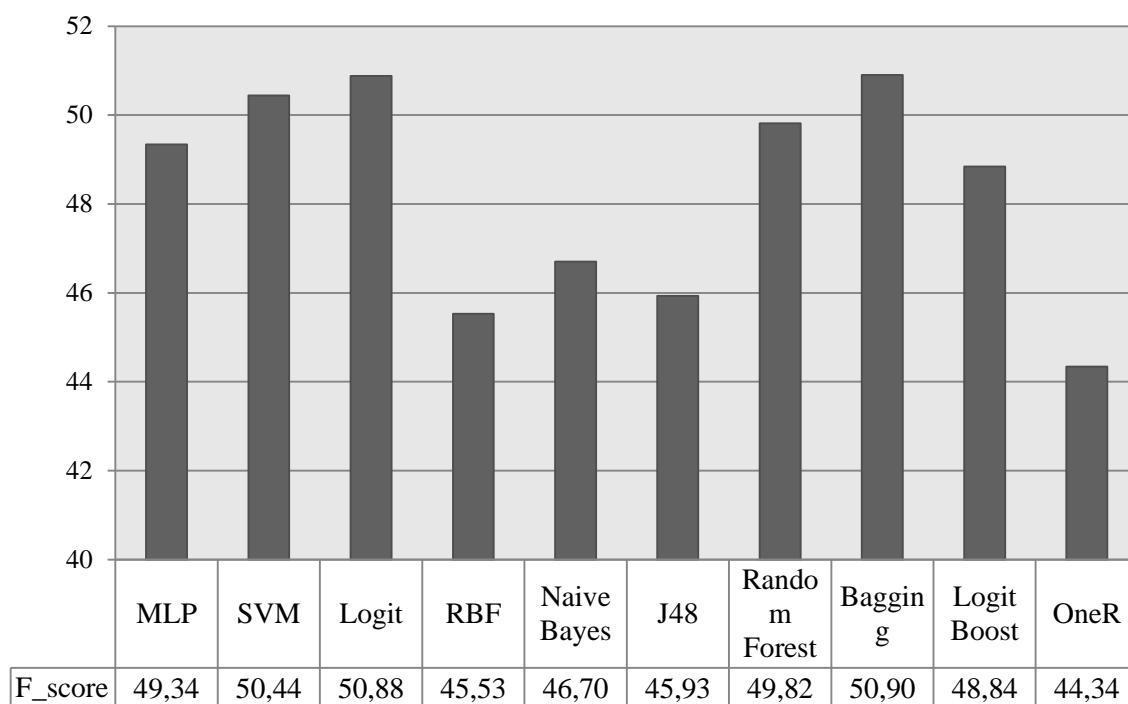


Рисунок 3.5 – Результаты работы классификаторов на БД *UUDB* (*F-score*, %)

Таблица 3.3 – Пример матрицы неточностей для базы данных *Emo-DB*, полученной с помощью машин опорных векторов

		Действительные значения							Точность в пределах класса, %
		Счастье	Нейтрально	Злость	Грусть	Страх	Скука	Отвращение	
Предсказанные	Счастье	49	2	14	0	10	0	2	63.64
	Нейтрально	1	70	0	0	0	9	0	87.50
	Злость	13	0	113	0	3	0	1	86.92
	Грусть	0	0	0	52	2	7	1	83.87
	Страх	7	1	0	1	53	1	2	81.54
	Скука	0	6	0	7	0	64	3	80.00
	Отвращение	1	0	0	2	1	0	43	91.49
Полнота в пределах класса, %		69.01	88.61	88.98	83.87	76.81	79.01	82.69	

На примере БД *EMO-DB* и классификатора *SVM* рассмотрим технологию вычисления метрики *F-score*. В таблице 3.3 представлена матрица неточностей, полученная по 6 прогонам в ходе процедуры кросс-валидации.

Точность и полнота системы в целом приняли следующие значения: $Precision = 82.14\%$, $Recall = 81.28\%$. Тогда показатель *F-score* равен:

$$F_score = 2 \cdot \frac{81.28 \cdot 82.14}{81.28 + 82.14} = 81.71\% \quad (3.4)$$

Подобным образом были вычислены значения показателя *F-score* в остальных экспериментах.

Анализ результатов, представленных на рисунках 3.1-3.4, показал, что на множестве рассматриваемых БД такие классификационные модели, как полносвязный персептрон, машины опорных векторов и логистическая регрессия демонстрируют наибольшую эффективность по сравнению с другими классификаторами. Поэтому было решено сосредоточиться на совершенствовании указанных моделей, в частности, далее рассматривается подход для автоматического генерирования полносвязных персептронов с учетом их вычислительной сложности (пункт 3.3), а также на объединении данных классификаторов в коллектив с целью повышения надежности (глава 4).

Перед представлением подхода, разработанного для проектирования структур нейронных сетей, приведем основные свойства данных моделей, виды нейросетевых классификаторов и алгоритмы их обучения.

3.2 Искусственные нейронные сети как инструмент интеллектуального анализа данных

В рамках теории искусственного интеллекта можно выделить обширный класс нейросетевых моделей, разработка и принцип функционирования которых были инспирированы попытками изучить природу человеческого мозга [10]. Существующая аналогия с биологическими нейронными сетями предопределила возможности данных вычислительных моделей: искусственные нейронные сети

(ИНС) демонстрируют способность к обучению на основе предыдущего опыта, к извлечению существенных свойств из поступающей информации, а также к обобщению имеющихся прецедентов на новые случаи [13].

Первые ИНС были спроектированы в 50-60ые годы XX века на основе биологических и физиологических исследований головного мозга. Минский (Minsky), Розенблатт (Rosenblatt), Уидроу (Widrow) разработали сети, состоящие из одного слоя искусственных нейронов [15, 16]. Часто называемые перцептронами, они были использованы для решения таких задач, как предсказание погоды, анализ электрокардиограмм и искусственное зрение. Однако из-за слабо проработанной теоретической базы проектирование нейросетевого алгоритмического аппарата было приостановлено на некоторое время. Позже ученые Кохонен (Kohonen), Гроссберг (Grossberg), Андерсон (Anderson), а также исследовательская группа по параллельным вычислениям (Rumelhart D.E., Hinton G.E., Williams R.J., США) продолжили исследования в области ИНС [107]. Постепенно появился теоретический фундамент, на основе которого сегодня конструируются наиболее мощные многослойные сети [7, 9].

Как уже было отмечено, идея ИНС была заимствована учеными из биологии: нейроны, складываясь в единую сеть, обеспечивают возможность параллельной обработки сигналов и информации в организме человека [5, 8].

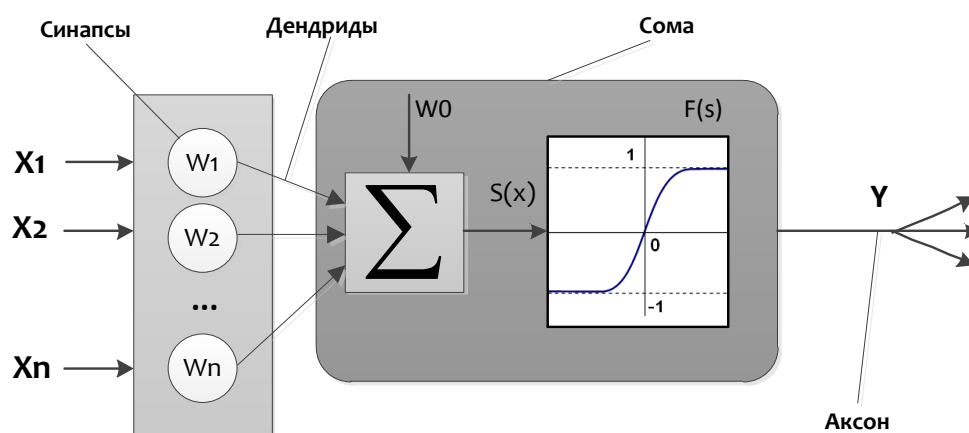


Рисунок 3.6 – Модель искусственного нейрона

Схема работы искусственного нейрона во многом схожа с функционированием его биологического прототипа. Отдельный нейрон имеет множество входов и лишь один выход. Входные значения взвешиваются, после чего суммируются. Далее полученный результат используется в качестве аргумента функции активации, выходное значение которой совпадает с выходом нейрона (рисунок 3.6).

Условно ИНС в целом можно определить как направленный граф, узлы (нейроны) которого соединены взвешенными связями [75]. По типу связей, присутствующих в архитектуре сети, ИНС можно классифицировать на:

- сети прямого распространения (без петель);
- рекуррентные сети (с обратными связями).

Первый класс составляют статические сети (выходные значения не зависят от предыдущих состояний), среди которых одно- и многослойные персептроны, а также сети радиально-базисных функций (РБФ) [65]. Ко второй группе относят динамические модели: соревновательные сети, сети Кохонена (Kohonen) и Хопфилда (Hopfield), а также модели адаптивно резонансной теории (ART).

В контексте ИНС обучение рассматривается как выбор эффективной структуры и настройка соответствующих весов для качественного решения задачи. Зачастую этот процесс является итеративным: с течением эпох функционирование ИНС постепенно улучшается. Выделяют три типа обучения [3]:

- с учителем (алгоритм настройки сети оперирует правильными ответами, выходами, для всех тренировочных примеров);
- без учителя (используется лишь информация о структуре тренировочных данных, корреляции между образами);
- смешанное (комбинированный подход: одна часть весов определяется в ходе самообучения, другая – в ходе обучения с учителем).

В теории представлено четыре основных типа правил, согласно которым осуществляется обучение [3]:

Правило коррекции по ошибке: коррекция весов происходит с учетом разницы между реальным и желаемым значениями выходов сети для каждого обучающего примера.

Обучение Больцмана: стохастическое правило обучения, при котором состояния нейронов должны удовлетворять желаемому распределению вероятностей.

Правило Хебба: особенностью данного правила является то, что изменение весовых коэффициентов зависит только от активности связанных с ними нейронов.

Обучение методом соревнования: нейроны выходного слоя соревнуются между собой за активацию («победитель берет все»).

Одним из самых популярных алгоритмов обучения ИНС остается алгоритм *обратного распространения ошибки*, реализующий настройку весов полносвязного многослойного персептрона на основе правила коррекции по ошибке [11, 12].

Алгоритм обратного распространения ошибки (Backpropagation)

Алгоритм обратного распространения ошибки был предложен в 1974 г. учеными А.И. Галушкиным и Полом Дж. Вербосом (Werbos P.J.), разработавшими его независимо друг от друга. Алгоритм основывается на итеративном градиентном спуске, в ходе которого осуществляется минимизация ошибки, определяемой как разница между выходом многослойного персептрона и его идеальным значением. Название алгоритма обусловлено распространением величины ошибки от выходов сети к входному слою нейронов, что соответствует обратному режиму работы для сетей прямой проводимости. Рассмотрим алгоритм подробнее [11].

Введем целевой критерий, отражающий ошибку ИНС:

$$E(w) = \frac{1}{2} \sum_{j,p} (y_{j,p}^{(N)} - d_{j,p})^2, \quad (3.5)$$

где $y_{j,p}^{(N)}$ – выход j -ого нейрона на выходном слое ИНС при подаче на ее входы p -ого объекта обучающей выборки, $d_{j,p}$ – идеальное значение выхода этого же нейрона.

Минимизация критерия (3.5) производится итеративно на основе метода градиентного спуска. Настройка весов сети реализуется через оценку корректирующей компоненты:

$$\Delta w_{ij}^{(n)} = -\eta \cdot \frac{\partial E}{\partial w_{ij}}, \quad (3.6)$$

где w_{ij} – значение весового коэффициента, соответствующего связи между i -ым нейроном слоя $n-1$ и j -ым нейроном слоя n , η – параметр скорости обучения, $\eta \in (0, 1)$.

Аналогично методу наименьших квадратов, найдем производную критерия (3.5) по w_{ij} :

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial y_j} \cdot \frac{dy_j}{ds_j} \cdot \frac{\partial s_j}{\partial w_{ij}}, \quad (3.7)$$

где y_j – по-прежнему, выходное значение j -ого нейрона, а $s_j = \sum_i w_{ij} \cdot x_i$ – аргумент активационной функции.

Множитель $\frac{dy_j}{ds_j}$ представляет производную активационной функции по ее аргументу, что накладывает на функцию условия непрерывной дифференцируемости. Множитель $\frac{\partial s_j}{\partial w_{ij}}$ совпадает с выходом нейрона предыдущего слоя $y_i^{(n-1)}$. В свою очередь, первый множитель можно представить как:

$$\frac{\partial E}{\partial y_j} = \sum_k \frac{\partial E}{\partial y_k} \cdot \frac{dy_k}{ds_k} \cdot \frac{\partial s_k}{\partial y_j} = \sum_k \frac{\partial E}{\partial y_k} \cdot \frac{dy_k}{ds_k} \cdot w_{jk}^{(n+1)}. \quad (3.8)$$

Суммирование по k ведется для нейронов слоя $n+1$. Обозначим:

$$\delta_j^{(n)} = \frac{\partial E}{\partial y_j} \cdot \frac{dy_j}{ds_j}. \quad (3.9)$$

В результате, имеем рекурсивную формулу для расчета $\delta_j^{(n)}$ n -ого слоя на основе значений $\delta_k^{(n+1)}$ слоя $n+1$:

$$\delta_j^{(n)} = \left[\sum_k \delta_k^{(n+1)} \cdot w_{jk}^{(n+1)} \right] \cdot \frac{dy_j}{ds_j}, \quad (3.10)$$

и для выходного слоя:

$$\delta_j^{(N)} = (y_l^{(N)} - d_l) \cdot \frac{dy_l}{ds_l}. \quad (3.11)$$

Перепишем (3.6) в новых обозначениях:

$$\Delta w_{ij}^{(n)} = -\eta \cdot \delta_j^{(n)} \cdot y_i^{(n-1)}. \quad (3.12)$$

Таким образом, алгоритм обратного распространения ошибки включает следующие этапы:

1. Рассчитать значения на выходном слое нейронов при подаче на входы ИНС примера из обучающей выборки.

2. Вычислить $\delta_j^{(N)}$ по формуле (3.11) для выходного слоя ИНС.

Рассчитать $\Delta w^{(N)}$ по формуле (3.12).

3. Определить $\delta_j^{(n)}$ и $\Delta w^{(n)}$ для слоев $n=N-1, \dots, 1$.

4. Скорректировать весовые коэффициенты:

$$w_{ij}^{(n)}(t) = w_{ij}^{(n)}(t-1) + \Delta w_{ij}^{(n)}(t). \quad (3.13)$$

5. Проверить критерий останова (число итераций, величина ошибки на обучающей выборке, изменение ошибки на последних итерациях): если не выполняется, то перейти к шагу 1.

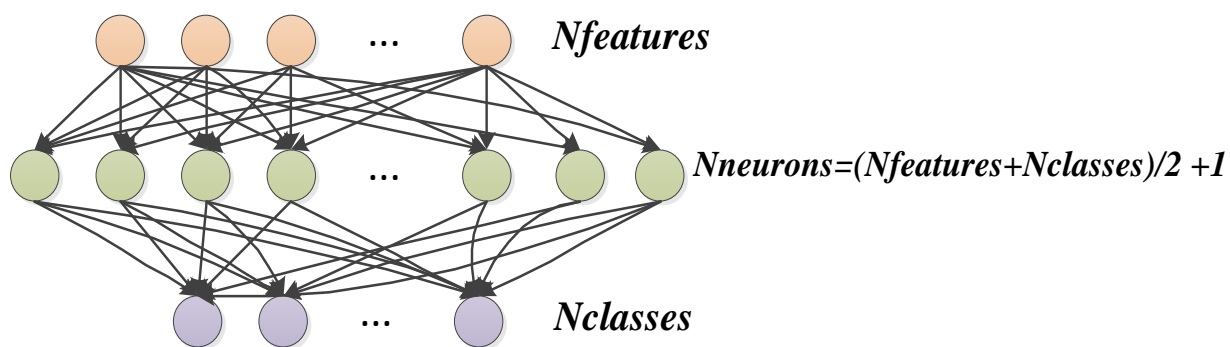


Рисунок 3.7 – Модель полносвязного персептрона с одним скрытым слоем

Обычно подача обучающих примеров на входы ИНС осуществляется в случайном порядке. После того, как все примеры были использованы для обучения, начинается новая *эпоха*. Тренировочные образы в новом порядке подаются на входы сети.

В качестве стандартной структуры полносвязного персептрона принята ИНС с одним скрытым слоем нейронов с активационными сигмоидальными функциями (рисунок 3.7).

В случае если число атрибутов или классов в выборке велико, то на скрытом слое также размещается большое количество нейронов, вследствие чего, растет и число связей в сети. Так, например, стандартная структура персептрона для задачи распознавания эмоций содержит 384 нейрона на входном слое и почти 200 нейронов на скрытом. В связи с большим количеством весовых коэффициентов требующих настройки, применение альтернативных методов, например, эволюционных или стайных алгоритмов, становится невозможным.

Широкое распространение ИНС получили благодаря следующим свойствам:

- обучаясь на тренировочных данных, сеть раскрывает неизвестные зависимости между входами и выходами;
- за счет настройки весовых коэффициентов ИНС способны автоматически определять информативные признаки, т.е. отпадает необходимость дополнительного анализа входных данных;

- адаптация к условиям функционирования (в том числе в режиме реального времени);
- быстродействие за счет параллелизма;
- надежность работы (достигается за счет распределенной конфигурации).

На сегодняшний день сфера применения ИНС включает в себя следующие предметные области:

- техника и телекоммуникации (оптимизация пути пересылки трафика между узлами) [127];
- информационные технологии (определение тематики текстовых сообщений) [132];
- экономика и финансы (предсказания курсов валют, предварительная обработка транзакций, прогнозирование задолженностей) [108];
- реклама и маркетинг (анализ рынка сбыта товаров, системы планирования цен и затрат) [83];
- здравоохранение (диагностика различного рода заболеваний) [23].

Внедрение ИНС в другие области – вопрос времени, т.к. развертывание интеллектуальной информационной технологии на практике является трудоемкой и сложной процедурой [2]. Главным вопросом остается выбор структуры сети и пути ее обучения для эффективного решения задачи. Однако даже для людей, владеющих экспертными знаниями предметной области, а также теорией ИНС, данный выбор представляется затруднительным. Поэтому учеными активно разрабатываются подходы, направленные на автоматизацию проектирования нейросетевых моделей. Нередко указанные подходы базируются на применении эвристических оптимизационных процедур таких, как эволюционные и стайные алгоритмы [51].

3.3 Автоматическое генерирование полносвязных перцептронов многокритериальным генетическим алгоритмом

Как было отмечено ранее, зачастую в программных пакетах, содержащих реализации алгоритмов интеллектуального анализа данных, структура нейронных сетей определяется однозначно по следующему правилу: на единственном скрытом слое число нейронов равно $(\frac{N_{features} + N_{classes}}{2} + 1)$, где $N_{features}$ – размерность входного вектора признаков, $N_{classes}$ – число различных классов в выборочных данных; тип активационной функции – сигмоид. Однако нередко такая структура является избыточной: число связей достаточно велико в связи с чем, на обучение модели и оценку выходных значений тратится значительное количество вычислительных ресурсов. Поэтому целесообразно использовать нейросетевые модели, отвечающие не только требованиям точности, но и вычислительной сложности.

Была разработана двухкритериальная оптимизационная модель задачи генерирования нейронных сетей, а именно полносвязных перцептронов с одним скрытым слоем. Пусть $\omega \in \Omega$, Ω – множество полносвязных перцептронов с различными скрытыми слоями. Введем критерии:

Критерий 1. Относительная ошибка классификации:

$$K1(\omega) = E(\omega) = \frac{N_{incorrectly}(\omega)}{N_{all}} \rightarrow \min_{\omega}, \quad (3.14)$$

где $N_{incorrectly}(\omega)$ – число примеров из выборки, классифицированных неверно перцептроном ω , N_{all} – общее количество примеров в выборке.

Обучение перцептронов с различными скрытыми слоями и оценка эффективности моделей по критерию $K1$ осуществляется с помощью метода обратного распространения ошибки.

Критерий 2. Вычислительная сложность:

Вычислительная сложность перцептрона зависит от количества связей и числа нейронов, составляющих его полносвязную структуру [20]. Поэтому

выразим данный критерий через время, затрачиваемое на обработку всех связей и вычисление активационных функций:

$$K2(\omega) = \sum_{i=1}^{N_{weights}(\omega)} T_i^{weight} + \sum_{j=1}^{N_{neurons}(\omega)} T_j^{act} \rightarrow \min_{\omega}, \quad (3.15)$$

где $N_{weights}(\omega)$ – число связей в оцениваемой структуре ω , $N_{neurons}(\omega)$ – число нейронов на скрытом и выходном слое модели ω , T_i^{weight} – временные затраты на обработку одной связи, T_j^{act} – время, затрачиваемое на вычисление активационной функции j -ого нейрона сети ω .

T_i^{weight} одинаково для всех связей, поэтому запишем:

$$K2(\omega) = N_{weights}(\omega) \cdot T^{weight} + \sum_{j=1}^{N_{neurons}(\omega)} T_j^{act} \rightarrow \min_{\omega}, \quad (3.16)$$

Величины T^{weight} и T_j^{act} зависят от характеристик ЭВМ, в связи с чем целесообразно перейти к безразмерным составляющим критерия $K2(\omega)$. Разделим обе части выражения (3.16) на T^{weight} :

$$\frac{K2(\omega)}{T^{weight}} = N_{weights}(\omega) + \sum_{j=1}^{N_{neurons}(\omega)} \frac{T_j^{act}}{T^{weight}} \rightarrow \min_{\omega}, \quad (3.17)$$

Введем обозначение:

$$K_j = \frac{T_j^{act}}{T^{weight}}, \quad (3.18)$$

где K_j – не зависящий от аппаратной реализации коэффициент относительной сложности вычисления активационной функции j -ого нейрона.

Получим выражение для критерия $K2$:

$$K2(\omega) = N_{weights}(\omega) + \sum_{j=1}^{N_{neurons}(\omega)} K_j \rightarrow \min_{\omega}, \quad (3.19)$$

где значения K_j определяются эмпирически на основании T_j^{act} и T^{weight} .

Для решения поставленной задачи предлагается использовать коллективный эволюционный алгоритм, оперирующий бинарными строками. Структура нейронной сети кодируется следующим образом (рисунок 3.8): хромосома содержит коды активационных функций (из конечного множества альтернативных вариантов) для каждого нейрона скрытого слоя, при этом нулевой код означает отсутствие нейрона.

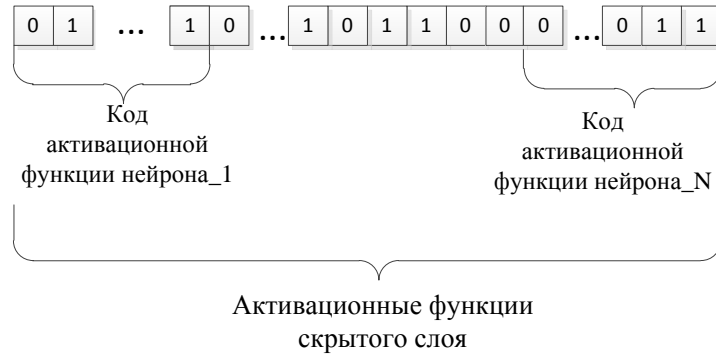


Рисунок 3.8 – Представление скрытого слоя персептрона в виде бинарной хромосомы

Входными параметрами являются множество активационных функций (с кодами) и максимальное число нейронов на скрытом слое (в проведенных экспериментах оно равно числу нейронов на скрытом слое персептрона со стандартной структурой). Активационные функции выходного слоя персептрона в данной хромосоме не кодируются. По умолчанию принято, что это сигмоиды.

Таблица 3.4 – Множество активационных функций

Код	Активационная функция	Значение K_i
1	$f(x) = \begin{cases} -1, & x < -1 \\ x, & -1 \leq x \leq 1 \\ 1, & x > 1 \end{cases}$	6.46
2	$f(x) = 1$	2.69
3	$f(x) = e^{-\frac{x^2}{2}}$	22.48
4	$f(x) = \tanh(x)$	23.14
5	$f(x) = \frac{1}{1 + e^{-x}}$	22.20
6	$f(x) = 1 - e^{-\frac{x^2}{2}}$	20.55
7	$f(x) = \operatorname{atan}(x)$	27.01

Таблица 3.4 содержит активационные функции, используемые для проектирования скрытого слоя ИНС. Коэффициенты $K_i = \frac{T_i^{act}}{T_{weight}}$ были оценены эмпирически, причем время обработки одной связи определялось как время, затрачиваемое на умножение, а затем сложение двух вещественных чисел (моделирование процесса добавления связи в структуру персептрона).

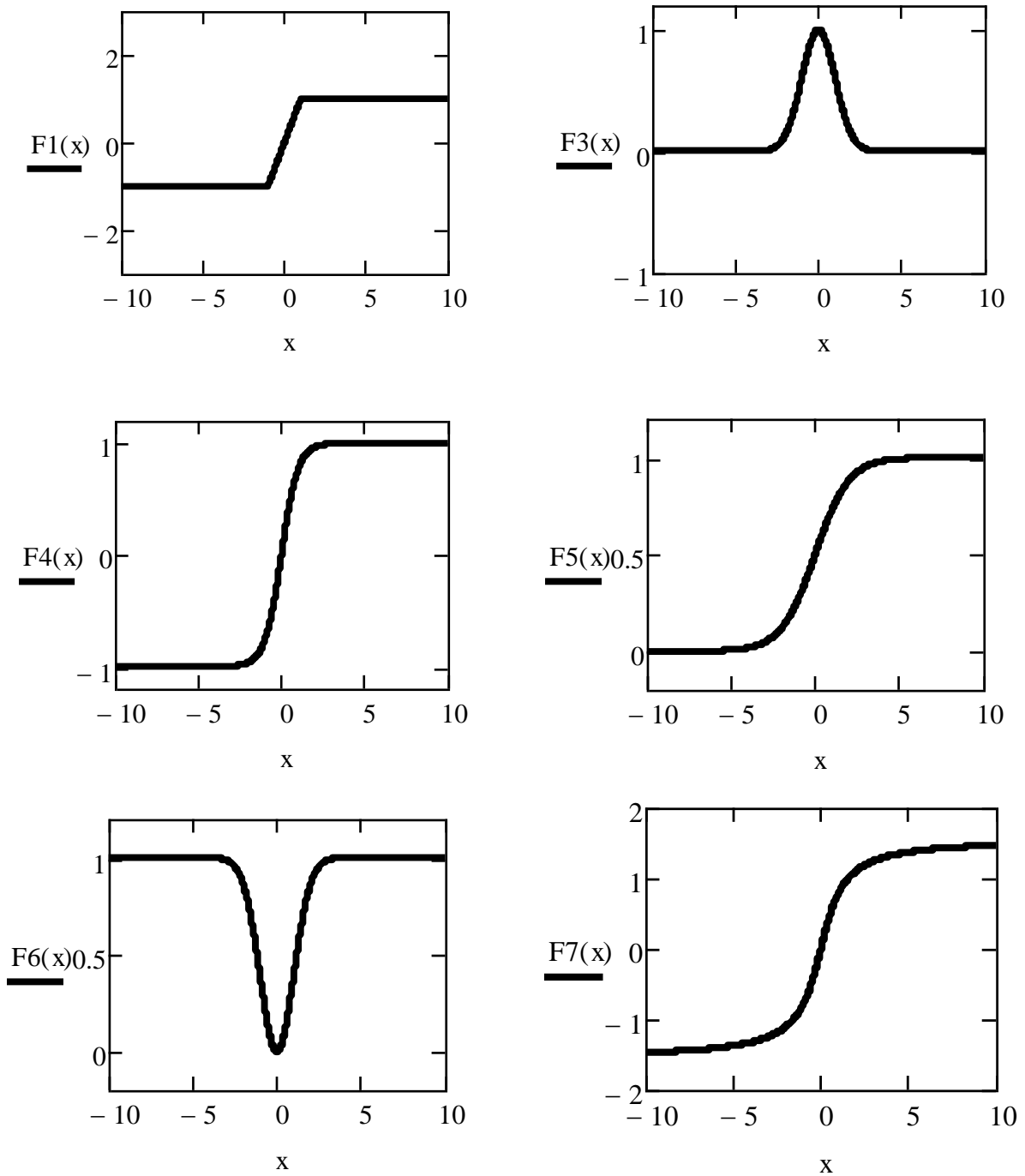


Рисунок 3.9 – Активационные функции

Как было отмечено, активационные функции должны отвечать требованию непрерывной дифференцируемости. Однако Функция 1 не удовлетворяет данному условию в точках $x=1$ и $x=-1$. В некоторых источниках говорится, что использование подобного рода функций недопустимо, в других же – лишь рекомендуется избегать точки, в которых производная не существует. В данной работе было введено допущение (аналогично делению на 0 в генетическом программировании), что производная Функции 1 в точках $x=1$ и $x=-1$ существует и равна 1.

На рисунке 3.9 представлены графики используемых активационных функций $F1(x)$, $F3(x)$ - $F7(x)$.

Заметим, что данные функции определены на всей числовой оси, а также ограничены сверху и снизу, что является их преимуществом при использовании в качестве функций активации. Кроме того, Функции 2-7 являются гладкими, т.е. непрерывно дифференцируемыми (допущения, введенные для Функции 1, описаны выше).

Поскольку при двухкритериальной постановке задачи выходом эволюционного алгоритма является набор недоминируемых решений, то у пользователя, во-первых, появляется возможность выбора «компромиссной» с его точки зрения модели (в смысле точности и вычислительной сложности), а, во-вторых, сгенерированные сети могут быть включены в ансамбль (коллектив) классификаторов для принятия коллективного решения.

Эффективность представленной многокритериальной эволюционной технологии генерирования полносвязных персептронов была исследована на задаче распознавания эмоций человека по речи. Далее приведены условия экспериментов, полученные результаты и основные выводы.

3.4 Исследование эффективности многокритериального генетического алгоритма автоматического проектирования полносвязных персептронов

В данном эксперименте, как и в предыдущем, была реализована процедура кросс-валидации с 6-тью стратифицированными частями. Результаты эксперимента вновь оценивались с помощью метрики F-score.

Известно, что выходом многокритериального ГА является множество несравнимых между собой точек – аппроксимация множества (фронта) Парето. Поэтому после завершения работы коллективного ГА необходимо из совокупности альтернативных структур (недоминируемых индивидов) выбрать финальное решение. С этой целью обучающая выборка случайным образом разделялась на 2 части (стратифицированно) – тренировочную и оценочную в пропорции 80% на 20% (рисунок 3.10).

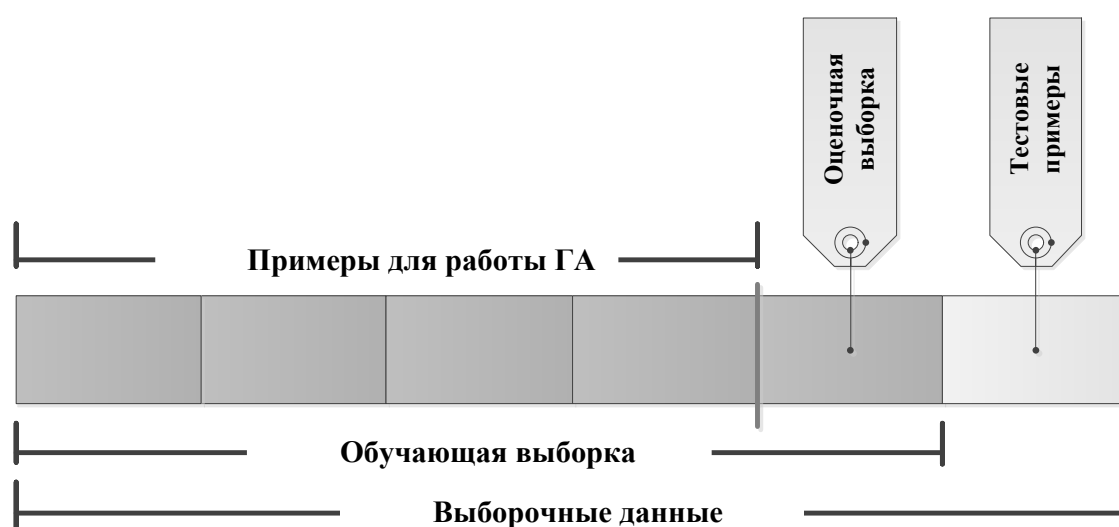


Рисунок 3.10 – Разбиение выборки для эксперимента

Соответственно, первая часть использовалась для определения пригодности кандидатов-решений в коллективном многокритериальном ГА, а вторая – для оценки эффективности полученных альтернативных структур по критерию «относительная ошибка классификации». Персептрон с наименьшей ошибкой на оценочной выборке принимался за финальное решение.

Параллельно функционирующим популяциям многокритериального ГА было выделено равное количество ресурсов: число поколений – 20 и объем популяции – $30/3=10$ индивидов. Размер миграции положили равным 3 (т.е. каждый «остров» получал в целом 6 кандидатов от двух других «островов»), интервал миграции – 5 поколений. Для каждой компоненты коллективного многокритериального ГА (NSGA-II, PICEA-g и SPEA2) были выбраны настройки генетических операторов: турнирная селекция (размер турнира – 2), равномерное скрещивание и вероятность мутации $p_m=1/n$, где n – длина хромосомы.

Для обучения персептронов с различными скрытыми слоями, т.е. для оценки индивидов по критерию «*относительная ошибка классификации*», применялся алгоритм обратного распространения ошибки (25 эпох, параметр $\eta=0.01$), причем для получения усредненного значения на данном этапе также применялась процедура кросс-валидации (число частей, на которые разделялась выборка, было равно 3).

На последнем поколении работы ГА популяции и внешние множества (архивы с недоминируемыми индивидами) всех «островов» объединялись в одну популяцию: общее количество кандидатов-решений было равно $3 \cdot 10 + 2 \cdot 7 = 44$ (10 – размер популяции на всех «островах» и 7 – размер внешнего множества для алгоритмов PICEA-g и SPEA2). Далее использовалась оценочная выборка для определения наиболее эффективной модели: каждая структура обучалась алгоритмом обратного распространения ошибки на тренировочных примерах, число эпох было равно 100, параметр $\eta=0.01$. По результатам классификации на оценочной выборке (значения критерия «*относительная ошибка классификации*») выбиралось финальное решение.

Затем обучающая и оценочная выборки объединялись в единое множество примеров для настройки полученной модели: на данном этапе число эпох в алгоритме обратного распространения ошибки было равно 250, $\eta=0.01$.

Таблица 3.5 – Результаты экспериментов

Название БД	Классификатор	Значение F-score	Значение критерия K2	Число нейронов на скрытом слое
Емо-DB	Классический перцептрон (WEKA)	80.83	81759.8	197
	Перцептрон, спроектированный коллективным ГА	80.75 (↓0.10%)	62492.6 (↑23.57%)	152.2
	Ансамбль перцептронов, спроектированных коллективным ГА	82.90 (↑2.56%)	62276.5 (↑23.83%)	151.6
SAVEE	Классический перцептрон (WEKA)	59.55	81759.8	197
	Перцептрон, спроектированный коллективным ГА	61.69 (↑3.59%)	62212.1 (↑23.91%)	151.5
	Ансамбль перцептронов, спроектированных коллективным ГА	62.02 (↑4.15%)	62677.2 (↑23.34%)	152.7
LEGO	Классический перцептрон (WEKA)	68.19	80058.6	195
	Перцептрон, спроектированный коллективным ГА	66.03 (↓3.17%)	62678.2 (↑21.71%)	154.3
	Ансамбль перцептронов, спроектированных коллективным ГА	71.05 (↑4.19%)	62172.6 (↑22.34%)	153.1
UUDB	Классический перцептрон (WEKA)	49.34	80276.8	195
	Перцептрон, спроектированный коллективным ГА	47.18 (↓4.38%)	62667.9 (↑21.94%)	154.0
	Ансамбль перцептронов, спроектированных коллективным ГА	50.68 (↑2.72%)	63033.0 (↑21.48%)	154.7

Таблица 3.5 содержит результаты проведенного эксперимента. Эффективность автоматически спроектированных моделей (строки таблицы,

обозначенные «*Перцептрон, спроектированный коллективным ГА*») сравнивалась с эффективностью стандартного перцептрона, реализованного в программной системе WEKA (строки таблицы «*Классический перцептрон (WEKA)*»). Стандартная структура полносвязного перцептрона также была оценена по критерию K2 – «Вычислительная сложность». В скобках указано относительное улучшение (или ухудшение) значений F-score и критерия K2, соответствующих автоматически сгенерированным ИНС, по сравнению с их классическими аналогами.

С помощью статистической проверки (t-критерия Стьюдента при уровне значимости $p=0.01$) было установлено, что ни одна из автоматически спроектированных нейросетевых моделей не уступает по эффективности (в смысле критерия F-score) полносвязному перцептрону со стандартной структурой. При этом полученные ИНС имеют более простую организацию, о чем свидетельствуют значения критерия «вычислительная сложность» (относительное улучшение для всех БД превышает 20%).

Однако наличие множества несравнимых решений на выходе ГА является и преимуществом, т.к. перцептроны с различной структурой могут быть включены в коллектив классификаторов для повышения качества получаемых решений. Так, в ходе эксперимента после получения финальной популяции, состоящей из 44 кандидатов-решений, формировался ансамбль нейросетевых моделей, включающий 15 наиболее эффективных перцептронов, которые были выявлены при помощи оценочной выборки.

Далее ансамбль классификаторов обучался на всех тренировочных примерах, включая оценочную выборку, алгоритмом обратного распространения ошибки (число эпох было равно 250, $\eta=0.01$). Финальное решение о принадлежности тестовых примеров к тому или иному классу принималось на основании голосования по правилу большинства.

Таблица 3.5 также содержит результаты данного эксперимента (строки, обозначенные «*Ансамбль перцептронов, спроектированных коллективным ГА*»). Значения критерия K2 и число нейронов на скрытом слое были усреднены не

только по 6 прогонам кросс-валидационной процедуры, но и по всем членам ансамбля. Относительное улучшение метрики F-score также оценивалось по сравнению с результатами работы полносвязного персептрона со стандартной структурой. Заметим, что для всех рассматриваемых БД применение ансамбля автоматически сгенерированных ИНС приводит к повышению качества получаемых решений (относительное улучшение метрики F-score варьируется от 2.56% (Emo-DB) до 4.19% (LEGO)).

Использование ансамбля спроектированных персептронов оправдано еще и потому, что не всегда применяемая модель одинаково эффективна на оценочной и тестовой выборках. Классификатор, демонстрирующий наилучший результат на оценочных примерах, может быть наихудшим на множестве тестовых образов. Поэтому для повышения надежности системы рекомендуется использовать ансамбль различных ИНС. Полученные результаты (таблица 3.4) показывают, что во всех проведенных экспериментах набор моделей превосходит по эффективности наилучшую сеть, выбранную по оценочным примерам.

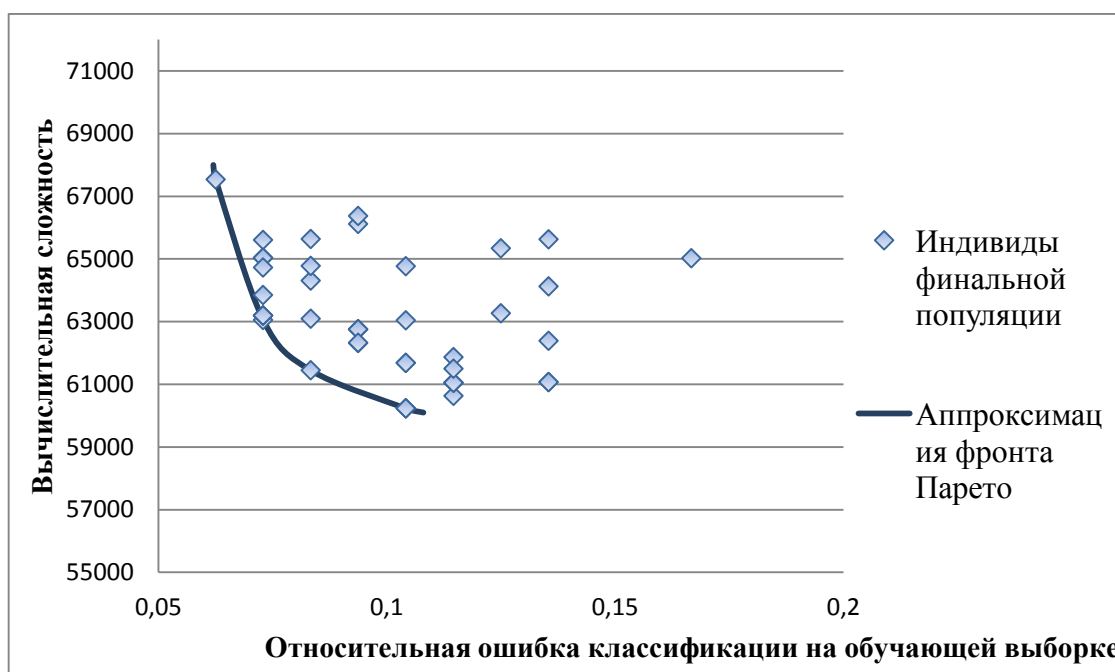


Рисунок 3.11 – Индивиды финальной популяции в критериальном пространстве и аппроксимация фронта Парето для базы данных Emo-DB

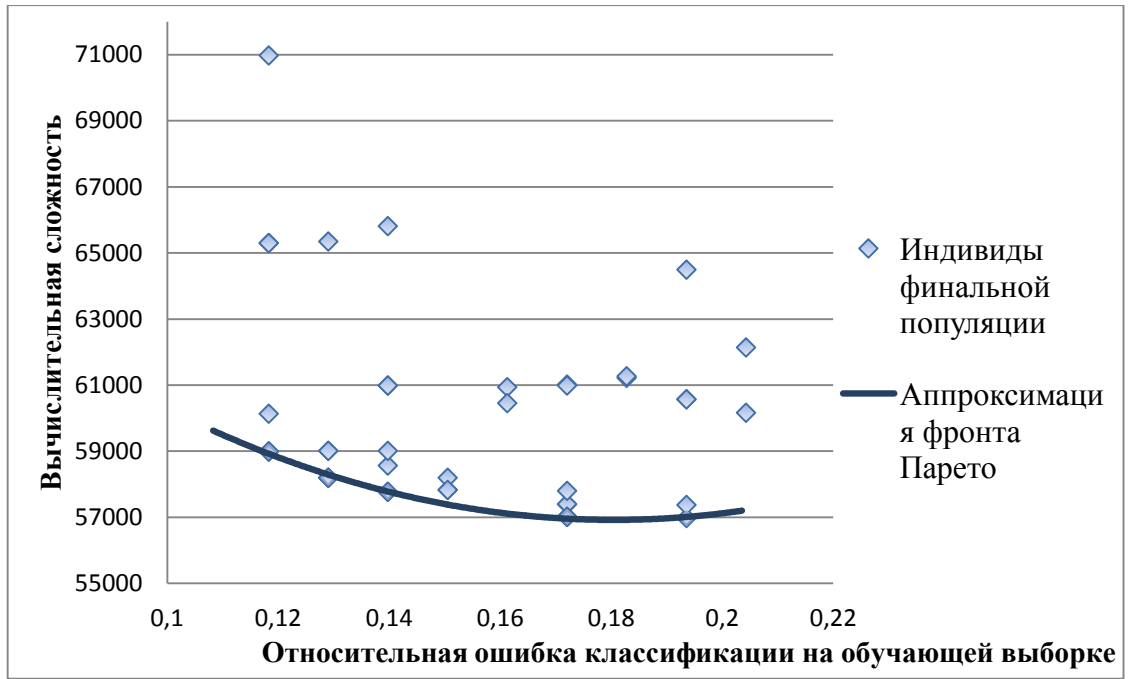


Рисунок 3.12 – Индивиды финальной популяции в критериальном пространстве и аппроксимация фронта Парето для базы данных SAVEE

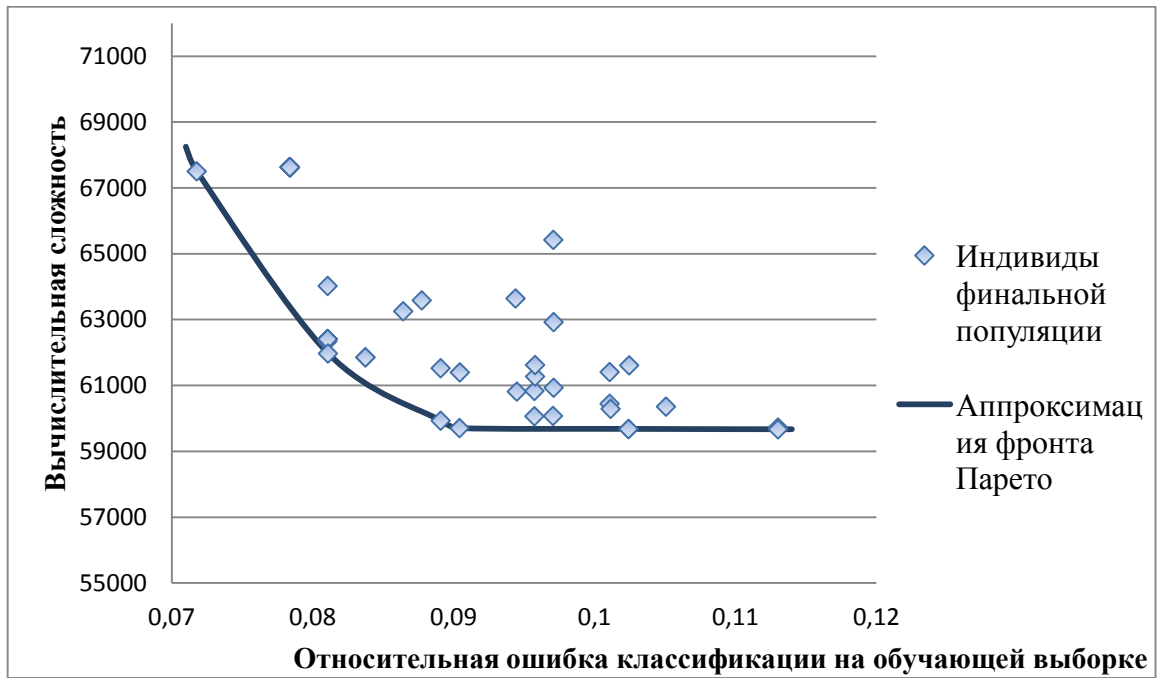


Рисунок 3.13 – Индивиды финальной популяции в критериальном пространстве и аппроксимация фронта Парето для базы данных LEGO

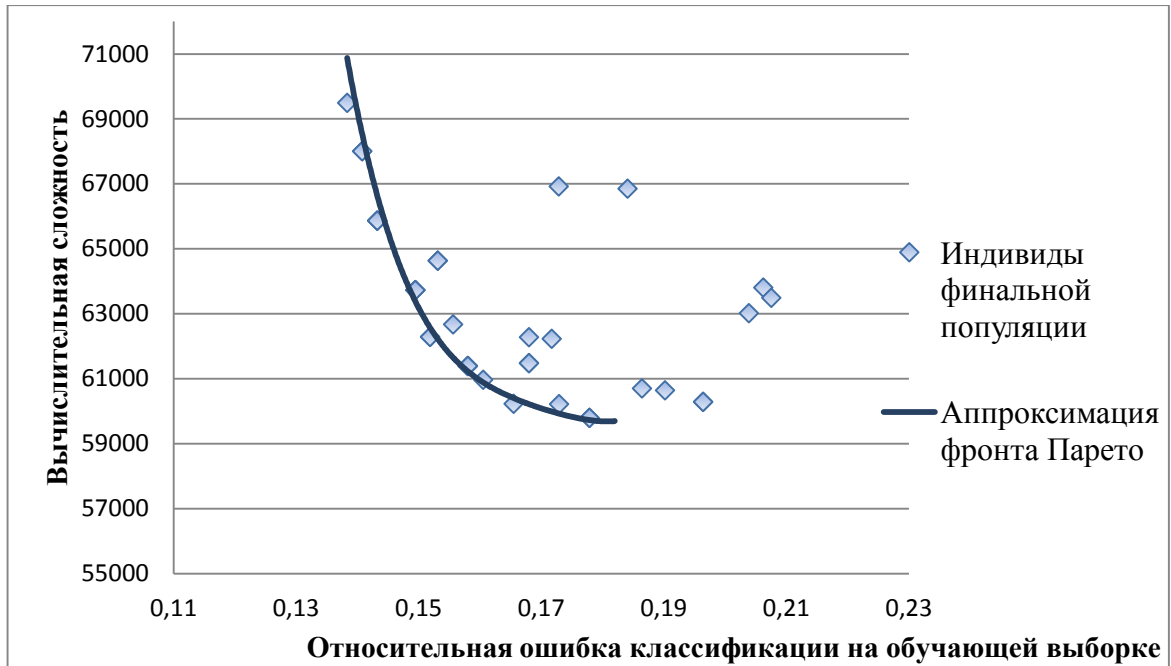


Рисунок 3.14 – Индивиды финальной популяции в критериальном пространстве и аппроксимация фронта Парето для базы данных UUDB

На рисунках 3.11-3.14 для каждой из рассматриваемых БД в критериальном пространстве изображены индивиды финальных популяций, полученные в ходе одного из прогонов. Во всех случаях можно обнаружить нелинейный характер аппроксимации фронта Парето.

В ходе анализа структур спроектированных полносвязных персептронов исследовался процент встречаемости каждой активационной функции на автоматически спроектированном скрытом слое. Для всех рассматриваемых БД по 6 прогонам кросс-валидационной процедуры на основании структур персептронов, включенных в ансамбль классификаторов, были получены средние значения встречаемости активационных функций (рисунки 3.15-3.18).

Для БД Eto-DB наиболее часто встречаемыми в структурах автоматически спроектированных ИНС были функции: $F2(x)$ – смещение, $F4(x)$ – гиперболический тангенс и $F7(x)$ – арктангенс. Причем сигмоидальная функция активации $F5(x)$, которая по умолчанию используется во всех прикладных программных пакетах, оказалась лишь четвертой в этом списке (рисунок 3.15).

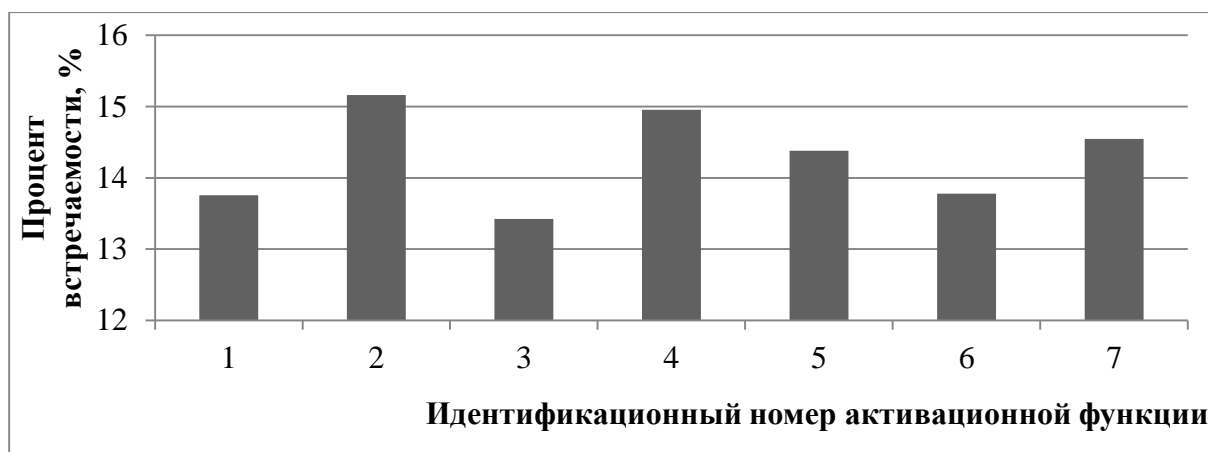


Рисунок 3.15 – Процент встречаемости активационных функций в структурах перцептронов для базы данных Emo-DB

На рисунке 3.16 приведены средние значения встречаемости активационных функций в структурах ИНС для БД SAVEE. Можно заметить, что двум самым простым (в смысле вычислительной сложности) функциям $F1(x)$ и $F2(x)$ соответствуют самые высокие показатели встречаемости. Сигмоид $F5(x)$ же занимает последнее место в данном списке.

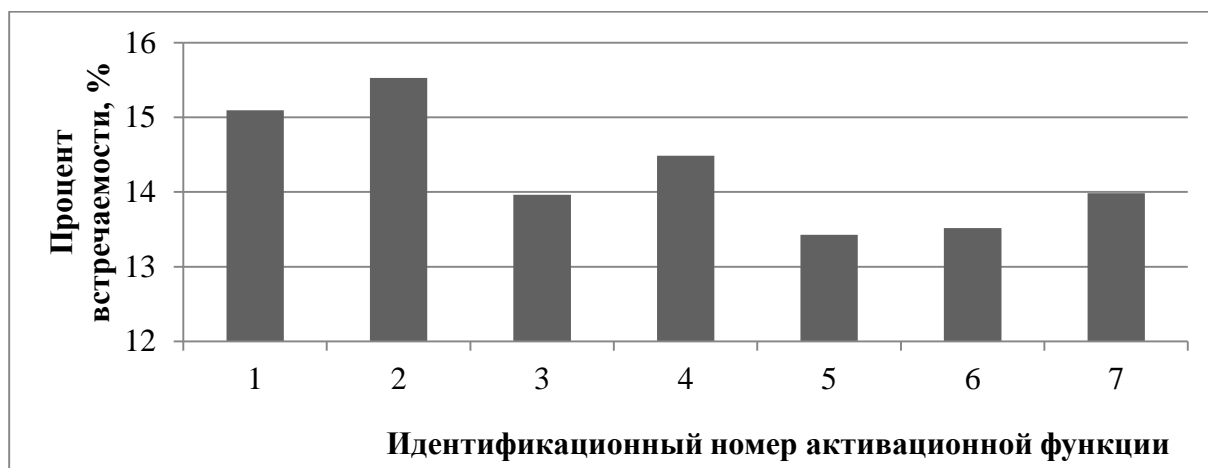


Рисунок 3.16 – Процент встречаемости активационных функций в структурах перцептронов для базы данных SAVEE

Для БД LEGO (рисунок 3.17) три самые часто используемые активационные функции: $F3(x)$ – экспоненциальная, $F4(x)$ – гиперболический тангенс и $F2(x)$ – смещение. Однако различие в средних значениях встречаемости с другими

функциями незначительно. В этом случае сигмоид $F5(x)$ также является редко используемой активационной функцией при проектировании скрытого слоя нейросетевых моделей.

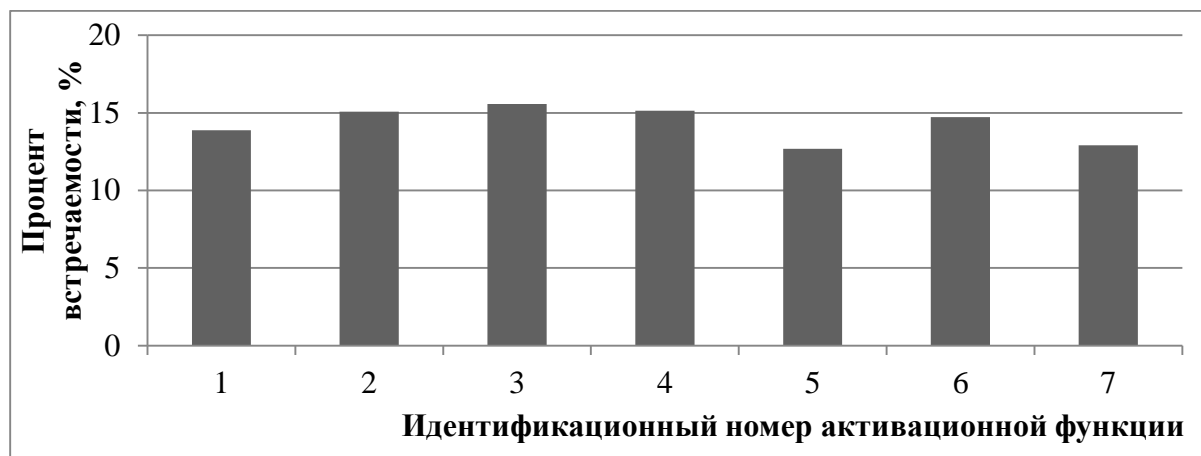


Рисунок 3.17 – Процент встречаемости активационных функций в структурах персептронов для базы данных LEGO

Было установлено, что для БД UUDV наибольший процент встречаемости имеют функции $F3(x)$ – экспоненциальная, $F4(x)$ – гиперболический тангенс и $F1(x)$ – кусочно-линейная (рисунок 3.18). Сигмоид $F5(x)$ лишь на четвертом месте.

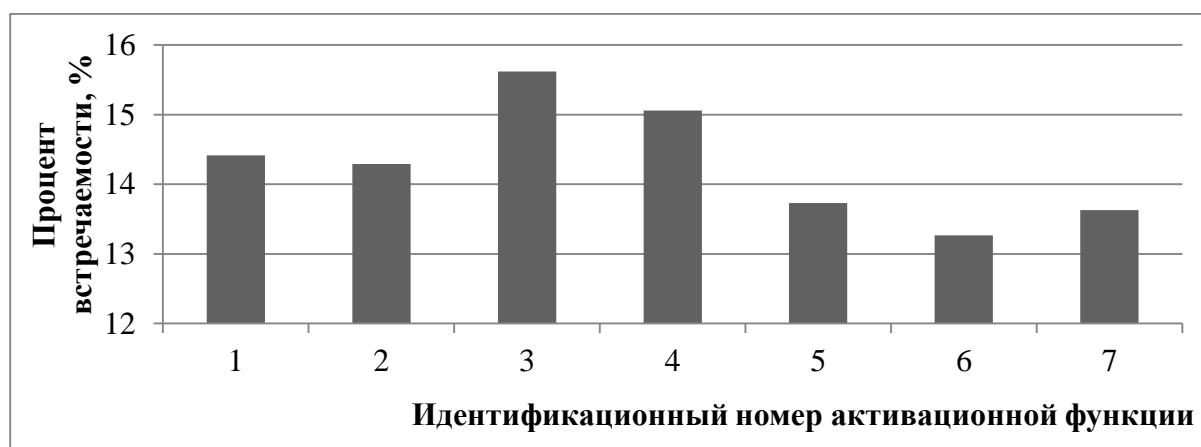


Рисунок 3.18 – Процент встречаемости активационных функций в структурах персептронов для базы данных UUDV

Анализ полученных результатов показал, что, во-первых, не существует набора активационных функций, частота использования которых для построения скрытого слоя нейронов одинакова для всех рассматриваемых БД. Другими словами, выбор подходящих активационных функций является нетривиальной задачей и должен осуществляться в автоматическом режиме с учетом особенностей задачи. Во-вторых, в ходе экспериментов было показано, что процент встречаемости сигмоидальной функции активации, которая применяется в большинстве случаев при реализации алгоритма обратного распространения ошибки, значительно снижается при возможности выбора альтернативных активационных функций. Кроме того, благодаря реализованному подходу было обнаружено, что, несмотря на свою простоту, функции $F1(x)$ и $F2(x)$ могут быть также эффективно использованы для формирования скрытого слоя персептрона, как и их более сложные аналоги (не зря в стандартную структуру ИНС, включающую только сигмоиды, всегда добавляют нейрон-смещение).

Таким образом, применение эвристической технологии проектирования нейронных сетей, а именно, полносвязных персептронов, на основе островного многокритериального ГА в задаче распознавания эмоций человека по речи позволяет генерировать модели, отличающиеся компактной, менее сложной (в вычислительном смысле) структурой по сравнению со стандартным персептроном, при этом, не снижая точности классификатора. Более того, многокритериальная модель задачи автоматического проектирования ИНС позволяет генерировать набор альтернативных моделей, которые могут быть включены в ансамбль нейросетевых классификаторов для повышения надежности системы распознавания.

Разработанный подход является универсальным в смысле решаемой задачи и может быть использован для проектирования нейросетевых моделей, предназначенных для классификации объектов, относящихся к любой предметной области.

ВЫВОДЫ

Полносвязный персептрон является одним из самых эффективных классификаторов, применяемых для распознавания эмоций человека по речи. Однако в силу свойств рассматриваемой задачи проектирование персептрона по общепринятому правилу порождает модель с избыточной структурой, что является недостатком при функционировании системы распознавания в режиме реального времени. Также вопросом, требующим решения, остается выбор подходящих активационных функций.

Учитывая данные особенности, был разработан подход для автоматического проектирования нейросетевых моделей, основанный на применении коллективного многокритериального генетического алгоритма. В рамках данного подхода генерирование полносвязных персептронов происходит с учетом дополнительного критерия «вычислительная сложность». В результате стало возможным формирование нейросетевых моделей, отличающихся компактной структурой, но не уступающих по точности стандартному персептрону. Дополнительными преимуществами являются автоматический выбор активационных функций, а также возможность проектирования ансамбля классификаторов для повышения надежности системы распознавания.

Кроме того, представление структуры ИНС в формате бинарной строки позволяет встроить процедуру отбора информативных признаков в процесс генерирования модели. В свою очередь, выбор релевантных атрибутов является актуальной задачей в области распознавания эмоций, поэтому в следующей главе рассмотрим преимущества данной технологии и сравним ее эффективность с другими алгоритмическими схемами.

Глава 4. Извлечение информативных признаков в задаче распознавания эмоций человека по речи коллективным генетическим алгоритмом многокритериальной оптимизации

4.1 Основные подходы к извлечению информативных признаков

Ранее было отмечено, что количество акустических характеристик, извлекаемых из голосового сигнала, достаточно велико и использование всего получаемого набора данных в процессе распознавания эмоций человека является нерациональным. Поэтому важной задачей в процессе идентификации эмоций говорящего является извлечение информативной подсистемы признаков, используемой алгоритмами распознавания.

Более того, при решении задач классификации целесообразно осуществлять предобработку данных, используемых алгоритмом обучения, еще и потому, что атрибуты могут иметь низкий уровень вариации, коррелировать друг с другом или содержать зашумленные измерения, снижающие точность классификатора.

Было показано [85], что если в выборке присутствуют атрибуты, являющиеся неинформативными для предсказания значений зависимой переменной, то точность деревьев решений, спроектированных алгоритмами ID3, C4.5, CART, значительно снижается, что справедливо и для метода k-ближайших соседей. Наивный байесовский классификатор обладает свойством робастности по отношению к нерелевантным атрибутам (его эффективность снижается очень медленно при добавлении неинформативных признаков). Однако при наличии коррелируемых атрибутов (пусть и релевантных) точность модели стремительно снижается.

Например, для задачи MONK1 [123] (база данных представляет собой характеристики робота) алгоритм C4.5 проектирует дерево с 15 внутренними узлами, 5 из которых содержат условия для проверки неинформативных признаков (в выборке их 3). При этом относительная ошибка классификации составляет 24.3%. Однако, если из выборки убрать нерелевантные атрибуты, то

ошибка снижается до 11.1%. Подобное поведение алгоритма C4.5 показано в [80] на задачах кредитного скоринга и диагностирования диабета.

В случае, когда стандартные методы извлечения информативных признаков (метод главных компонент, факторный анализ) не демонстрируют приемлемой эффективности, для решения данной задачи могут быть применены алгоритмические схемы, основанные на эвристических процедурах оптимизации.

В работе [85] описаны основные алгоритмические схемы, в рамках которых осуществляется отбор признаков из баз данных.



Рисунок 4.1 – Схема подхода *wrapper*

Подход *wrapper* – это комбинация оптимизационного алгоритма и классификатора, использующегося для оценки качества извлеченной подсистемы признаков (рисунок 4.1). Эффективность всех сокращенных наборов атрибутов,

рассматриваемых в качестве возможного решения задачи отбора признаков, может быть выражена несколькими критериями, но одним из них обязательно должен быть функционал отражающий качество классификации на данной подсистеме признаков (относительная ошибка/точность классификации, F-score и т.д.)



Рисунок 4.2 – Схема подхода *filter*

Извлечение атрибутов с помощью подхода *filter* основано на оценке статистических характеристик наборов данных (рисунок 4.2), т.е. ни один из введенных критериев не отражает качество работы классификационной модели на рассматриваемых подсистемах признаков. С одной стороны, игнорирование информации о релевантности извлеченного набора атрибутов для применяемого классификатора может снизить эффективность последнего. Однако данный подход требует меньше вычислительных ресурсов, т.к. отсутствует необходимость в обучении классификационной модели для оценки релевантности каждой рассматриваемой подсистемы признаков. Кроме того, подход *filter* может

быть применен для предобработки данных перед использованием коллектива классификаторов различной природы.

Учеными Venkatadri и Srinivasa (2010 г.) были описаны критерии, отражающие качество извлеченной системы атрибутов [126]:

1) Коэффициент несостоятельности (*англ.* The Inconsistent Example Pairs (IP) measure):

$$IP = \frac{N_{inconsistent_pairs}}{N_{pairs}}, \quad (4.1)$$

где $N_{inconsistent_pairs}$ – количество несостоятельных пар объектов в выборке, принадлежащих разным классам, N_{pairs} – общее количество пар объектов в выборке, принадлежащих разным классам.

Состоятельность пары объектов, относящихся к разным классам, означает отсутствие одинаковых значений для одного и того же признака.

2) Корреляция признаков (*англ.* Attribute Class Correlation (AC)):

$$AC = \frac{\sum w_i C(i)}{\sum w_i}, \quad (4.2)$$

где $C(i) = \frac{\sum_{j_1 \neq j_2} \|x_{j_1}(i) - x_{j_2}(i)\| \varphi(x_{j_1}, x_{j_2})}{n(n-1)/2}$, $x_j(i)$ – значение i -ого признака j -ого объекта выборки, $i = 1, \dots, m$, m – количество признаков, $j = 1, \dots, n$, n – объем выборки. $w_i = 1$, если признак i выбран; $w_i = 0$, если признак i не выбран. $\varphi(\dots, \dots) = 1$, если объекты j_1 и j_2 принадлежат разным классам; $\varphi(\dots, \dots) = 0.05$, если объекты j_1 и j_2 принадлежат одному классу. $\|\dots\|$ – обозначает модуль.

3) Внутрикласовое расстояние (*англ.* the Intra-Class Distance (IA)):

$$IA = \frac{1}{n} \sum_{r=1}^k \sum_{j=1}^{n_r} d(p_j^r, p_r), \quad (4.3)$$

где p_j^r – j -ый объект класса r , p_r – центральный объект класса r , n_r – число объектов в классе r , n – объем выборки, $d(\dots, \dots)$ – Евклидово расстояние.

4) Межкласовое расстояние (*англ.* the Inter-Class Distance (IE)):

$$IE = \frac{1}{n} \sum_{r=1}^k n_r d(p_r, p), \quad (4.4)$$

где p_r – центральный объект класса r , n_r – число объектов в классе r , p – центральный объект выборки, n – объем выборки, $d(\dots, \dots)$ – Евклидово расстояние.

5) Оценка Лапласа (*англ.* the Laplacian Score (LS)):

$$LS(i) = \frac{\tilde{x}(i)^T L \tilde{x}(i)}{\tilde{x}(i)^T D \tilde{x}(i)}, \quad (4.5)$$

где $\tilde{x} = x(i) - \frac{x(i)^T D(i)1}{1^T D(i)1}$, $x(i) = [x_1(i), x_2(i), \dots, x_n(i)]^T$, $1 = [1, \dots, 1]^T$, D и L – матрицы: $D = \text{diag}(S1)$, $L = D - S$, S – матрица коэффициентов, отражающих длину ребер в графе ближайших соседей.

6) Энтропия (*англ.* Representation Entropy(RE)):

$$RE = - \sum \tilde{\lambda}_i \log \tilde{\lambda}_i, \quad (4.6)$$

где $\tilde{\lambda}_i = \frac{\lambda_i}{\sum \lambda_i}$, λ_i – собственное значение ковариационной матрицы, соответствующее i -ому признаку.

Причем *критерий (4.1)* характеризуют подсистему признаков с позиций состоятельности атрибутов, *критерий (4.2)* с точки зрения их зависимости, *критерии (4.3) – (4.5)* основаны на вычислении расстояний, а *критерий (4.6)* является мерой информации. Совокупность целевых функционалов в задаче отбора признаков может быть представлена любой комбинацией данных критериев.

После построения математической модели необходимо выбрать алгоритм поиска оптимального решения. Простота представления подсистемы извлеченных атрибутов в виде бинарной строки, где 1 соответствует информативному признаку, а 0 – неинформативному, возможность поиска в дискретном пространстве с учетом нескольких критериев качества послужили главными причинами интереса исследователей, проявленного к эволюционным алгоритмам как инструменту решения данного класса задач.

В работах [131, 137] однокритериальный ГА используется в рамках схемы *wrapper*, где в качестве классификаторов выступают нейросетевая модель и

машины опорных векторов соответственно (единственный критерий точность классификации). ГА также применяется для минимизации критерия несостоятельности в схеме *filter*, реализованной в исследовании [91] для повышения эффективности алгоритма S4.5 и снижения вычислительных затрат.

Однако с развитием многокритериальных эволюционных алгоритмов стал возможен учет нескольких функционалов качества для оценки подсистемы выбранных атрибутов. В работе [126] исследуется эффективность различных комбинаций представленных ранее критериев (4.1) – (4.6), в качестве алгоритма поиска используется NSGA-II (схема *filter*). Этот же многокритериальный ГА применяется в [68] для нахождения компромиссного решения, где в качестве критериев выступают «число извлеченных признаков» и «точность классификатора k-ближайших соседей (k=1)» (схема *wrapper*).

В данной работе для извлечения релевантных атрибутов в задаче распознавания эмоций человека по речи были спроектированы многокритериальные модели, основанные как на схеме *wrapper*, так и на схеме *filter*. Предложенный ранее коллективный ГА был использован в качестве оптимизационного алгоритма. Подход, основанный на схеме *filter*, применялся в совокупности с ансамблем стандартных классификаторов различной природы, в то время как подход, базирующийся на схеме *wrapper*, был реализован в рамках эволюционной технологии автоматического генерирования перцептронов. Далее рассматриваются детали разработанных подходов.

4.2 Исследование эффективности подхода *filter* на задаче распознавания эмоций человека по речи

Перед реализацией и тестированием методов отбора информативных признаков, основанных на применении эволюционных алгоритмов, был проведен дополнительный эксперимент, подтверждающий необходимость их разработки.

Среди стандартных процедур снижения размерности одним из самых распространенных является метод главных компонент (*англ.* Principle Component

Analysis, PCA) [1]. Он был применен для задачи распознавания эмоций человека по речи с целью сокращения числа атрибутов и повышения эффективности классификационных моделей. В таблице 4.1 приведены результаты данного эксперимента для двух случаев: доля объясненной дисперсии была равна 0.75 и 0.95.

Таблица 4.1 – Результаты исследования метода главных компонент

Название БД	Метод отбора признаков	Число признаков	F-score, %		
			MLP	SVM	LOGIT
Емо-DB	Без PCA	384.00	80.83	81.71	80.46
	PCA (0.75)	49.67	74.71	79.61	77.04
	PCA (0.95)	136.80	73.87	73.62	76.39
SAVEE	Без PCA	384.00	59.55	59.31	60.82
	PCA (0.75)	46.67	57.46	57.86	59.86
	PCA (0.95)	130.7	50.63	46.18	51.80
LEGO	Без PCA	384.00	68.19	71.08	70.71
	PCA (0.75)	59.83	67.19	68.05	69.03
	PCA (0.95)	162.50	66.08	70.06	70.58
UUDB	Без PCA	384.00	49.34	50.44	50.88
	PCA (0.75)	46.67	47.53	48.48	49.61
	PCA (0.95)	156.80	47.93	49.37	49.89

В главе 3 были выявлены стандартные классификаторы, демонстрирующие наибольшую эффективность при распознавании эмоций. Полносвязный перцептрон, машины опорных векторов и логистическая регрессия использовались для сравнения точности распознавания до и после применения метода главных компонент. При постановке эксперимента была реализована кросс-валидационная проверка с 6 стратифицированными частями.

Анализ полученных результатов показал, что ни в одном из двенадцати случаев (по три классификатора для каждой из четырех баз данных) применение метода главных компонент не приводит к повышению эффективности

классификаторов, что обосновывает необходимость проектирования альтернативных подходов.

Рассмотрим разработанную эволюционную технологию отбора информативных признаков. Для извлечения релевантных атрибутов по схеме *filter* была использована двухкритериальная оптимизационная модель. Функционалами, описывающими качество извлеченной подсистемы информативных признаков, выбраны внутри- (*англ.* Intra-class distance (IA)) и межклассовое расстояния (*англ.* Inter-class distance (IE)):

$$\left\{ \begin{array}{l} IA(\delta) = \frac{1}{n} \sum_{r=1}^k \sum_{j=1}^{n_r} d(p_j^r(\delta), p_r(\delta)) \rightarrow \min_{\delta}, \\ IE(\delta) = \frac{1}{n} \sum_{r=1}^k n_r d(p_r(\delta), p(\delta)) \rightarrow \max_{\delta}, \end{array} \right. \quad (4.7)$$

где $\delta \in \Delta$, Δ – множество подсистем извлеченных атрибутов, $p_j^r(\delta)$ – j -ый экземпляр класса r в сокращенном наборе признаков δ , $p(\delta)$ – центр набора данных δ , $d(\dots, \dots)$ обозначает евклидову метрику, $p_r(\delta)$ и n_r соответствуют центру класса r и числу экземпляров в нем, k – количество классов в выборке, n – объем выборки. Для точек $p_r(\delta)$ и $p(\delta)$ i -ая координата ($i=1, \dots, m$, m – размерность признакового пространства) рассчитывается по формулам:

$$p_r^{(i)}(\delta) = \frac{1}{n_r} \sum_{j=1}^{n_r} p_j^{r(i)}(\delta), \quad p^{(i)}(\delta) = \frac{1}{n} \sum_{r=1}^k \sum_{j=1}^{n_r} p_j^{r(i)}(\delta). \quad (4.8)$$

В качестве алгоритма поиска оптимального решения целесообразно использовать многокритериальный ГА, оперирующий бинарными строками, где 1 кодирует информативный признак, 0 – неинформативный.

Поскольку выходом многокритериального ГА является множество недоминируемых по Парето решений, то в результате оптимизации критериев (4.7) имеем совокупность сокращенных БД. Далее каждая из полученных подсистем признаков используется для обучения модели, что приводит к формированию коллектива классификаторов. Финальное решение о

принадлежности объекта к тому или иному классу принимается путем голосования по правилу большинства (рисунок 4.3).

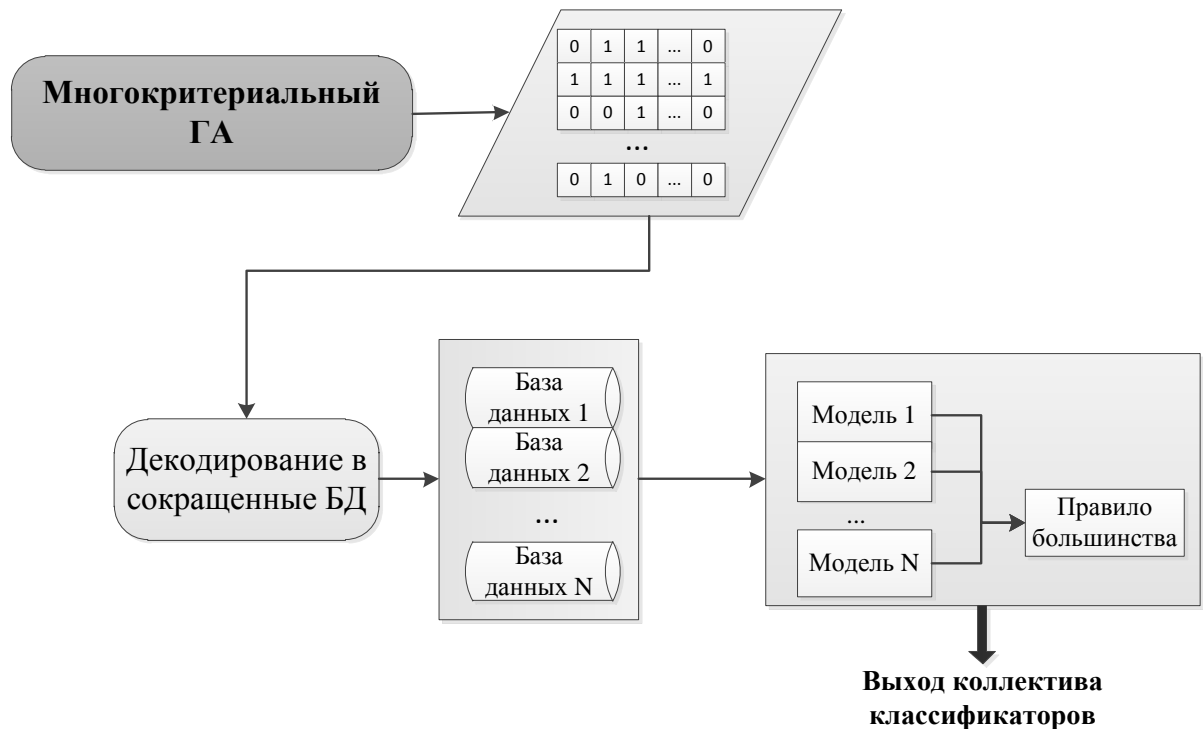


Рисунок 4.3 – Эволюционная технология извлечения информативных признаков по схеме filter с формированием коллектива классификаторов

Эксперимент 1. В первом эксперименте была исследована эффективность модели (4.7), положенной в основу эволюционной технологии извлечения информативных признаков в задаче распознавания эмоций человека по речи.

В качестве многокритериальных ГА были привлечены алгоритмы NSGA-II, PICEA-g, SPEA2, реализующие различные эвристики. Каждому из них было выделено равное количество ресурсов: 90 поколений и 150 индивидов в популяции. Размер множества недоминируемых решений – 30 (число сокращенных подсистем признаков). ГА запускались со следующими настройками: турнирная селекция (размер турнира – 2), равномерное скрещивание и вероятность мутации $p_m = 1/n$, где n – длина хромосомы.

Таблица 4.2 – Результаты Эксперимента 1

Название БД	Метод отбора признаков	Число признаков	F-score, %		
			MLP	SVM	LOGIT
Емо-DB	Без отбора признаков	384.00	80.83	81.71	80.46
	NSGA-II	165.16	86.07	82.77	82.56
	PICEA-g	180.88	84.74	83.06	84.09
	SPEA2	159.49	85.69	82.93	85.64
SAVEE	Без отбора признаков	384.00	59.55	59.31	60.82
	NSGA-II	163.67	67.70	66.61	69.84
	PICEA-g	186.06	68.81	64.80	64.28
	SPEA2	166.74	67.55	64.48	66.82
LEGO	Без отбора признаков	384.00	68.19	71.08	70.71
	NSGA-II	145.21	71.91	70.45	70.47
	PICEA-g	166.48	71.98	70.47	70.22
	SPEA2	151.18	72.71	70.61	70.58
UUDB	Без отбора признаков	384.00	49.34	50.44	50.88
	NSGA-II	141.47	50.44	50.12	50.50
	PICEA-g	167.50	50.22	50.34	50.91
	SPEA2	145.68	50.19	50.37	50.59

После завершения работы многокритериального ГА, т.е. на стадии обучения моделей, в качестве классификаторов использовались:

- полносвязный персептрон;
- машины опорных векторов;
- логистическая регрессия.

Как и в предыдущих экспериментах, была реализована процедура кросс-валидации с 6 стратифицированными частями. Качество классификации оценивалось метрикой F-score. Результаты эксперимента представлены в таблице 4.2. Количество извлеченных признаков, указанное для каждого эксперимента, усреднено по всему множеству сокращенных БД. Можно обнаружить, что применение разработанного подхода приводит:

- к повышению эффективности классификаторов в превалирующем числе случаев, причем в некоторых из них улучшение значительное: с 80.83% до 86.07% (БД Emo-DB, модель MLP – полносвязный персептрон) или с 60.82% до 69.84% (БД SAVEE, модель LOGIT – логистическая регрессия);
- к незначительному ухудшению в ряде случаев, которое статистически незначимо (выявлено с помощью t-критерия Стьюдента при уровне значимости $p=0.05$).

Заметим, что для БД Emo-DB и LEGO классификаторы, демонстрирующие лучший результат на полном наборе признаков (SVM), уступают по эффективности полносвязным персептронам (MLP) на сокращенном наборе признаков. Это значит, что невозможно заранее предугадать, какая из моделей будет наиболее эффективной при переходе к подсистеме извлеченных атрибутов. Кроме того, для различных БД «победителями» по значению метрики F-score среди многокритериальных ГА стали разные алгоритмы.

Проблема выбора наиболее эффективного классификатора может быть легко решена путем использования коллектива моделей MLP, SVM и LOGIT. Ведь главным преимуществом схемы *filter* является возможность ее применения в совокупности с ансамблем классификаторов различной природы.

В свою очередь, при переходе к коллективному островному ГА, включающему различные эвристики, отпадает и проблема выбора многокритериального ГА.

Эксперимент 2. В данном эксперименте для оптимизации системы критериев (4.7) был применен островной многокритериальный ГА, представленный в главе 2. Сочетание трех различных алгоритмов в одном позволяет обойти выбор наиболее эффективного ГА.

После завершения работы коллективного алгоритма вместо формирования ансамбля классификаторов одного типа происходит генерирование трех коллективов: полносвязных персептронов, машин опорных векторов и логистической регрессии (рисунок 4.4). Путем голосования внутри каждого

коллектива формируется решение о принадлежности объекта к какому-либо классу. Далее полученные коллективные решения участвуют в определении финального предсказания (по правилу большинства или случайным образом, если несколько классов имеет равное число голосов). Причем данный подход может быть расширен и на большее количество классификаторов, однако в работе выбраны именно указанные три модели как самые эффективные для распознавания эмоций человека по речи (опубликовано в [34]).

При запуске коллективного многокритериального ГА параллельно функционирующим популяциям были выделены ресурсы: 90 поколений и $150/3=50$ индивидов в популяции. Размер миграции приняли равным 10, интервал миграции – 10. Результирующая аппроксимация множества Парето содержала 30 решений (т.е. число сокращенных БД было равно 30). Для генетических операторов были использованы прежние настройки.

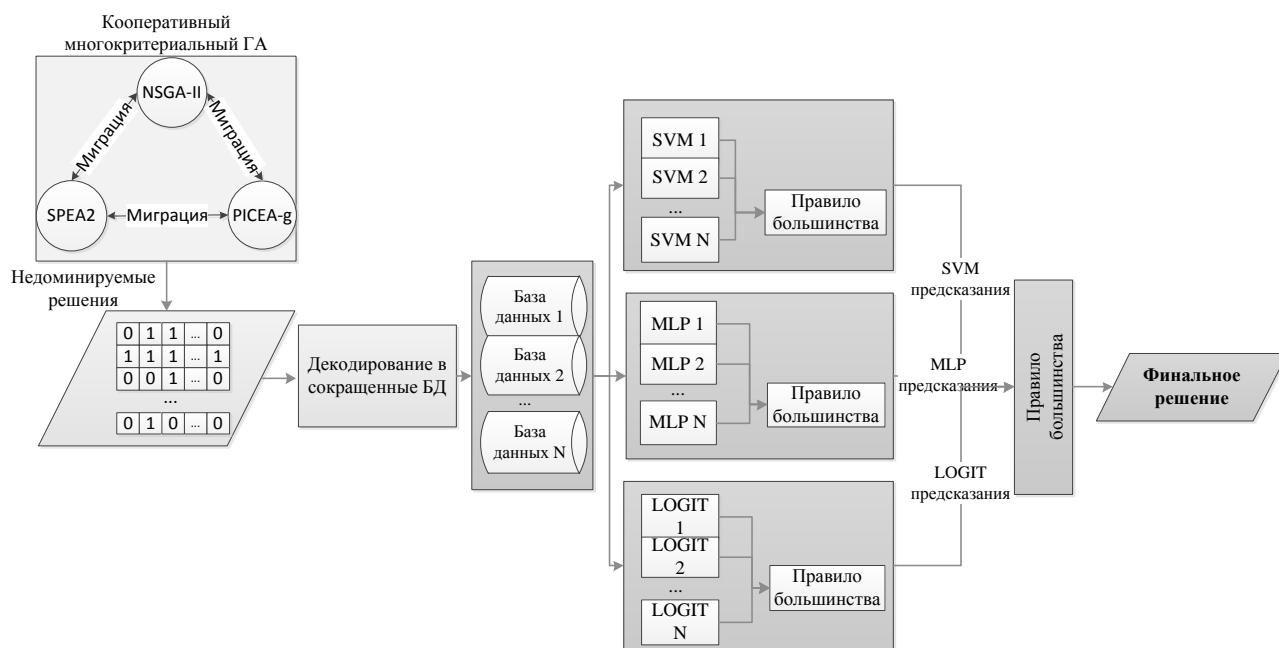


Рисунок 4.4 – Эволюционная технология извлечения информативных признаков с использованием нескольких классификационных моделей

Усредненные результаты, полученные в ходе кросс-валидационной проверки, приведены в таблице 4.3. Отметим, что в данном эксперименте полносвязные перцептроны проектировались по стандартной схеме, т.к. число

нейронов на скрытом слое сети было уменьшено за счет сокращения размерности входного вектора. В экспериментах 1 и 2 использовались реализации алгоритмов классификации из программной системы WEKA.

Таблица 4.3 – Результаты Эксперимента 2

Название БД	Метод отбора признаков	Число признаков	F-score, %		
			MLP	SVM	LOGIT
Емо-DB	Без отбора признаков	384.0	80.83	81.71	80.46
	Островной многокритериальный ГА	166.9	86.26 (↑5.57%)		
SAVEE	Без отбора признаков	384.0	59.55	59.31	60.82
	Островной многокритериальный ГА	165.9	68.71 (↑12.97%)		
LEGO	Без отбора признаков	384.0	68.19	71.08	70.71
	Островной многокритериальный ГА	150.7	71.29 (↑0.30%)		
UUDB	Без отбора признаков	384.0	49.34	50.44	50.88
	Островной многокритериальный ГА	146.9	51.02 (↑0.28%)		

В таблице 4.3 также представлено относительное улучшение метрики F-score по сравнению с наилучшим значением на полном наборе признаков для каждой из рассматриваемых БД (процент в скобках).

Кроме того, исследование результатов Эксперимента 2 включало и статистическую проверку. С помощью t-критерия Стьюдента (при уровне значимости $p=0.01$) было выявлено, что различия между значениями метрики F-score, полученными в результате применения коллективного многокритериального ГА и коллектива классификаторов, и значениями той же метрики, достигнутыми при помощи лучшей комбинации многокритериального ГА и классификационной модели (таблица 4.2), статистически не значимы. Следовательно, применение данного подхода позволяет избежать

дополнительных экспериментов для выбора наиболее эффективного ГА и классификатора.

Также заметим, что среднее количество атрибутов в сокращенных БД, полученных после завершения работы островного ГА, более чем в два раза меньше первоначальной размерности входного вектора признаков.

Таким образом, с помощью подхода, основанного на схеме filter, для Emo-DB и SAVEE удалось значительно превзойти точность распознавания наиболее эффективного классификатора на полном наборе признаков – на 5.57 и 12.97% соответственно. Применение данного подхода к БД LEGO и UUDB привело к сохранению точности распознавания, достигнутой лучшей классификационной моделью на всей совокупности атрибутов.

На рисунках 4.5-4.8 для каждой из рассматриваемых БД в критериальном пространстве изображены индивиды финальных популяций, полученные в ходе одного из прогонов.

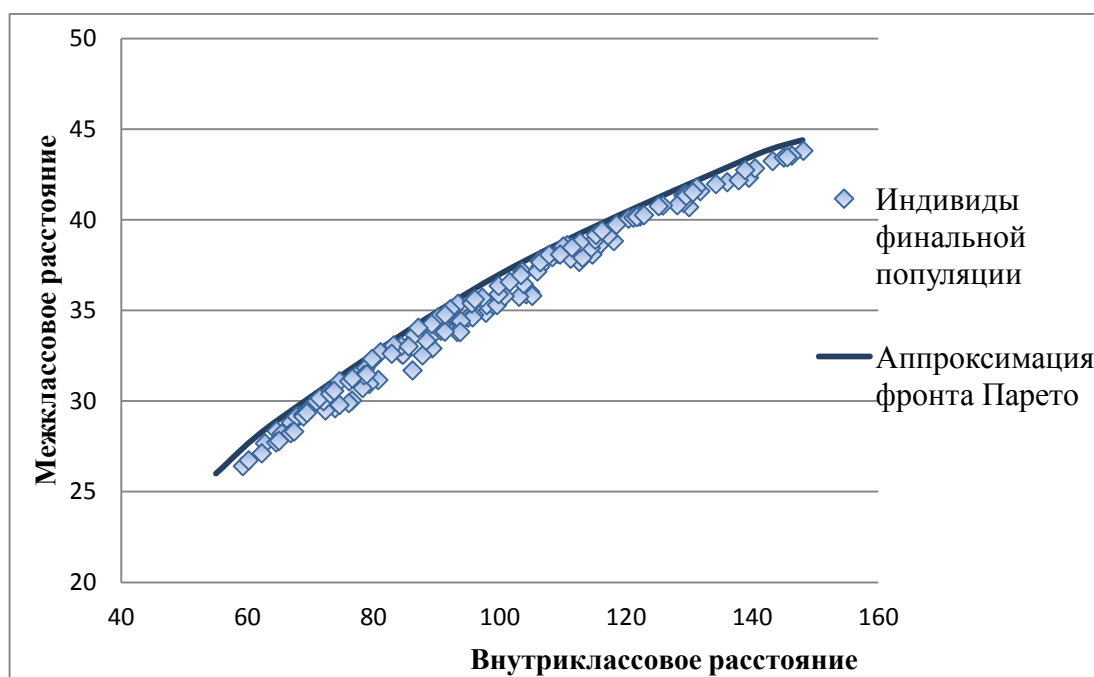


Рисунок 4.5 – Индивиды финальной популяции в критериальном пространстве и аппроксимация фронта Парето для базы данных Emo-DB

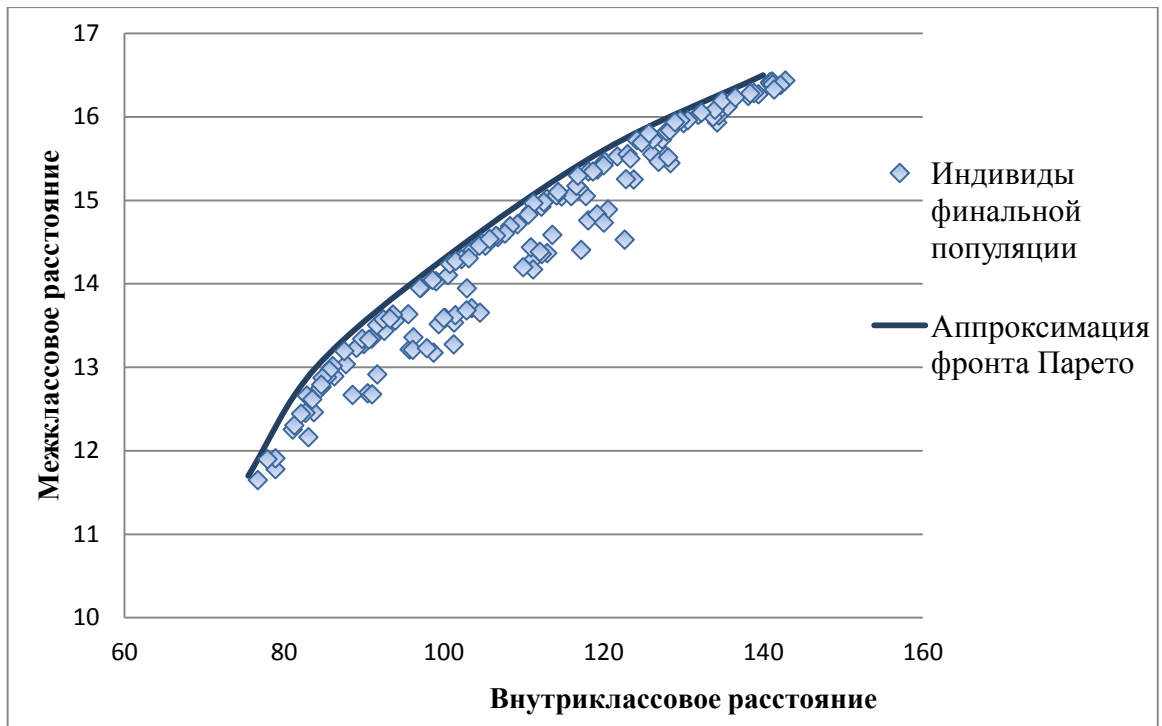


Рисунок 4.6 – Индивиды финальной популяции в критериальном пространстве и аппроксимация фронта Парето для базы данных SAVEE

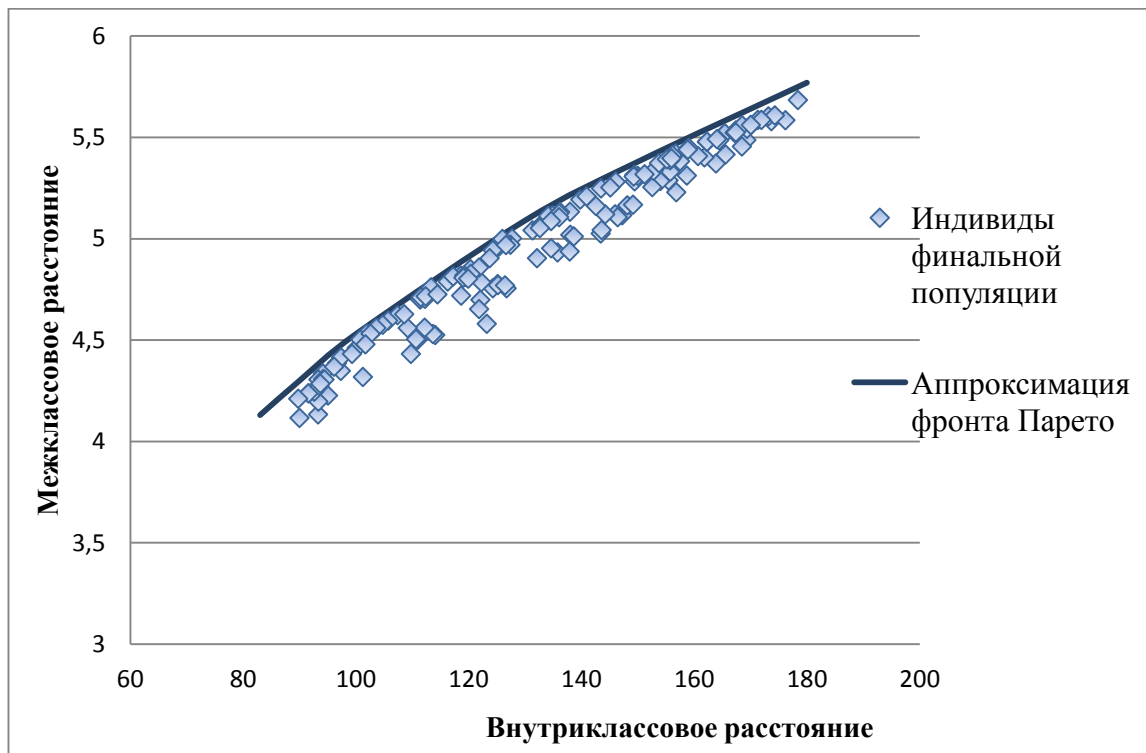


Рисунок 4.7 – Индивиды финальной популяции в критериальном пространстве и аппроксимация фронта Парето для базы данных LEGO

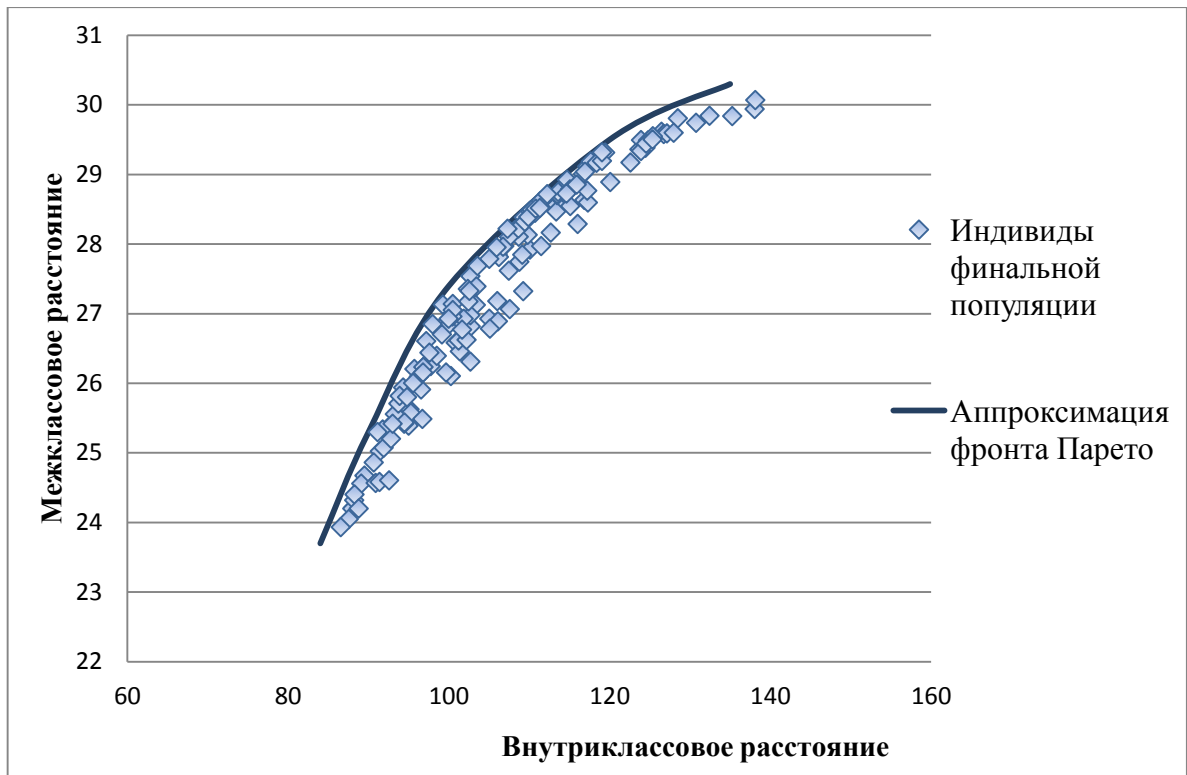


Рисунок 4.8 – Индивиды финальной популяции в критериальном пространстве и аппроксимация фронта Парето для базы данных UUDB

В большинстве случаев можно обнаружить близкий к линейному характер аппроксимации фронта Парето. Судя по значениям критериев, можно готовить о сильном смещении (перекрытии) примеров разных классов, поскольку среднее межклассовое расстояние меньше среднего внутриклассового.

Использование островного многокритериального ГА позволяет не только избежать выбора наиболее эффективного алгоритма для рассматриваемой задачи, но и сократить время, затрачиваемое его на работу. В среднем для всех БД параллельное функционирование «островов» привело к уменьшению временных затрат в 2.55 раза (с учетом времени, затрачиваемого на процесс миграции), что особенно важно для LEGO и UUDB в связи с их размерностью (рисунки 4.9–4.12).

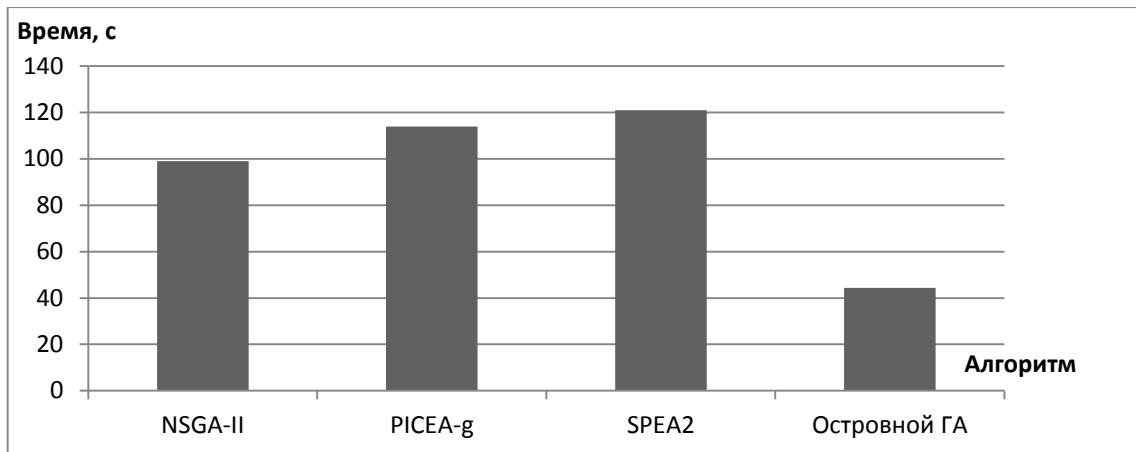


Рисунок 4.9 – Среднее время работы ГА для базы данных Eto-DB

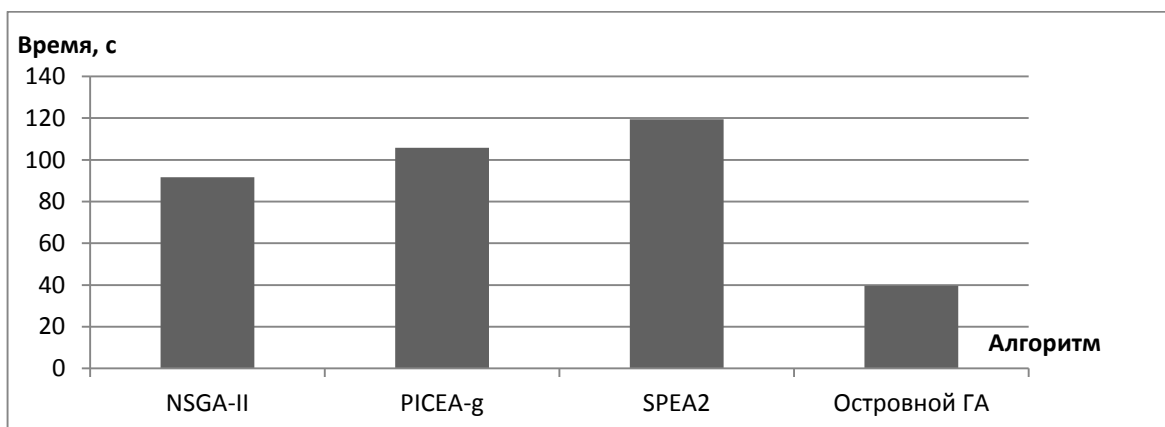


Рисунок 4.10 – Среднее время работы ГА для базы данных SAVEE

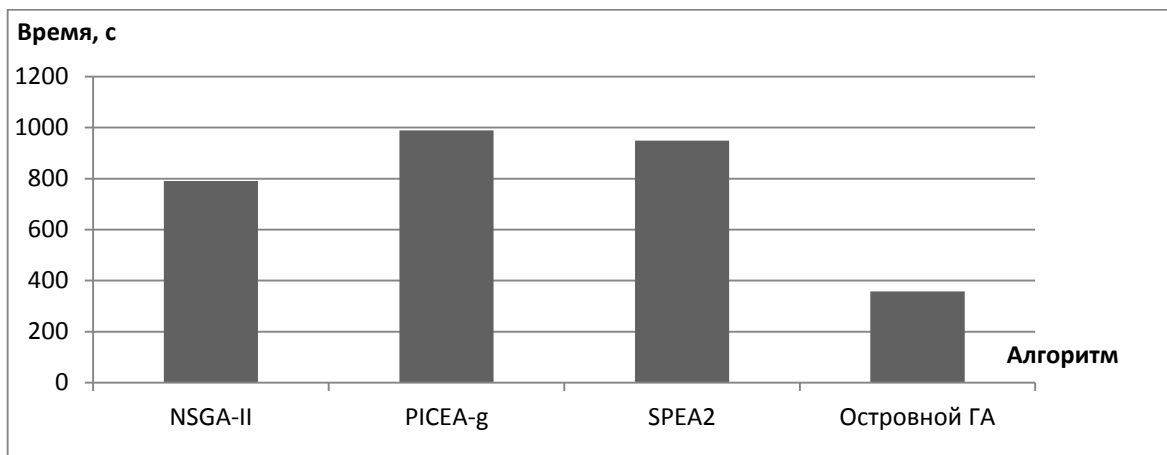


Рисунок 4.11 – Среднее время работы ГА для базы данных LEGO

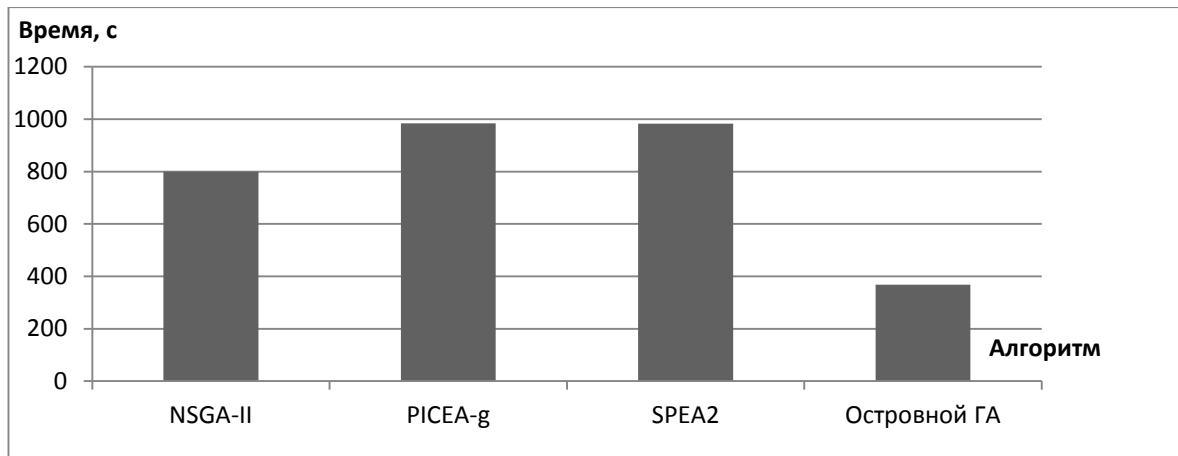


Рисунок 4.12 – Среднее время работы ГА для базы данных UUDB

Среди стандартных многокритериальных ГА NSGA-II является наиболее эффективным в смысле временного ресурса, затрачиваемого на его работу, т.е. на отбор признаков. Интересно, что на тестовых задачах, представленных в главе 2, было показано, что меньшее количество временных затрат требуется алгоритму PICEA-g (а не NSGA-II). Данный пример в очередной раз иллюстрирует невозможность экстраполяции результатов об эффективности ГА с одних задач на другие.

4.3 Исследование эффективности подхода *wrapper* на задаче распознавания эмоций человека по речи

Представленная в главе 3 эволюционная технология проектирования полносвязных перцептронов может быть дополнена процедурой автоматического извлечения информативных признаков.

Как было упомянуто ранее, модель задачи отбора релевантных атрибутов в рамках подхода *wrapper* обязательно включает критерий, отражающий качество классификации. Кроме того, она может содержать и ряд других критериев (например, число информативных признаков), в зависимости от требований, предъявляемых к подсистеме извлеченных атрибутов.

Напомним, что технология генерирования ИНС с помощью островного многокритериального ГА базируется на оптимизации двух критериев. Именно критерий K1 «*относительная ошибка классификации*» представляет собой функционал, наличие которого обязательно для схемы *wrapper*. В свою очередь, критерий K2 «*вычислительная сложность*» может быть использован для минимизации числа извлеченных атрибутов.

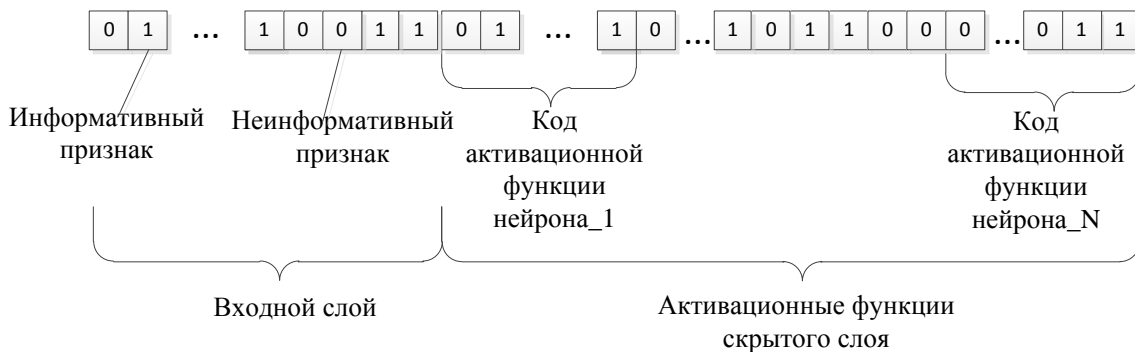


Рисунок 4.13 – Представление входного и скрытого слоя персептрона в виде бинарной хромосомы

Для этого входной слой нейронов также кодируется в бинарной строке: 1 соответствует информативному признаку, а 0 – неинформативному (рисунок 4.13). При этом слагаемое $N_{weights}$, учитываемое в K2 и обозначающее количество связей в сети, напрямую отражает число нейронов входного слоя, а значит, и размерность вектора признаков.

Для исследования эффективности подхода *wrapper*, встроенного в процедуру проектирования полносвязных персептронов, был повторен эксперимент, описанный в главе 3. Напомним, что была реализована кросс-валидационная проверка с 6-тью стратифицированными частями. В качестве основного показателя, отражающего качество классификации, использовалась метрика F-score. Обучающая выборка также делилась на тренировочную и оценочную в соотношении 80% на 20% (для определения финального решения из множества индивидов, недоминируемых по Парето). Для оценки бинарных строк

по критерию K1 вновь применялся алгоритм обратного ошибки ($\eta = 0.01$). Множество активационных функций представлено в таблице 3.4.

Поскольку длина хромосомы была увеличена на 384 бита, то в этом эксперименте островному многокритериальному ГА было выделено большее количество ресурсов: число поколений – 30 и объем каждой из трех популяций – $60/3=20$ индивидов. Размер миграции приняли равным 5, интервал миграции – 5 поколений. Для каждой компоненты коллективного многокритериального ГА (NSGA-II, PICEA-g и SPEA2) были выбраны прежние настройки генетических операторов: турнирная селекция (размер турнира – 2), равномерное скрещивание и вероятность мутации $p_m = 1/n$, где n – длина хромосомы.

Обычно при генерировании начальной популяции в ГА вероятности появления 0 и 1 полагают равными 0.5. Но для части хромосомы, отвечающей за вектор признаков (входной слой ИНС), целесообразно перераспределить вероятности в пользу появления 1 (0.8 для 1, 0.2 для 0), чтобы число извлеченных атрибутов сокращалось постепенно в процессе поиска.

В ходе работы ГА на обучение персептронов с различными структурами алгоритму обратного распространения ошибки выделялось 25 эпох (ошибка также усреднялась по 3 прогонам кросс-валидационной процедуры). При определении финального решения на оценочной выборке (среди недоминируемых альтернатив) число эпох было увеличено до 100. Наилучшая из моделей обучалась на всех тренировочных и оценочных примерах в течение 250 эпох.

В таблице 4.4 содержатся результаты проведенного эксперимента. Эффективность спроектированных ИНС с автоматическим отбором информативных признаков (строки таблицы обозначены «Персептрон, спроектированный коллективным ГА, со встроенной процедурой отбора информативных признаков») также сравнивалась с эффективностью стандартного персептрона, реализованного в программной системе WEKA (строки таблицы «Классический персептрон (WEKA)»).

Анализ полученных результатов показал, что использование спроектированных ИНС со встроенным отбором признаков не приводит к

значительному ухудшению или улучшению качества распознавания по сравнению со стандартным персептроном. Данное заключение было подтверждено результатами статистической проверки (t-критерий Стьюдента при уровне значимости $p=0.05$). Однако при сравнении вычислительной сложности моделей можно обнаружить, что для автоматически сгенерированных персептронов значение критерия K_2 почти в два раза ниже, что свидетельствует о значительном упрощении структуры. Это объясняется тем, что для всех рассматриваемых БД входной вектор признаков был сокращен более чем на 100 атрибутов. Поэтому с позиций вычислительных затрат спроектированные персептроны превосходят по эффективности стандартные аналоги.

Аналогично эксперименту в главе 3, была исследована эффективность проектирования ансамбля нейронных сетей с автоматически сгенерированным скрытым слоем. Из множества недоминируемых решений выбиралось 15 моделей, демонстрирующих наименьшую относительную ошибку классификации на оценочной выборке. Коллектив персептронов путем голосования формировал финальное решение, следуя правилу большинства. В таблице 4.4 строки *«Ансамбль персептронов, спроектированных коллективным ГА, со встроенной процедурой отбора информативных признаков»* содержат результаты данного эксперимента. Причем значения критерия K_2 , число нейронов на скрытом слое и размерность входного вектора признаков усреднены по всем членам коллектива.

По сравнению с точностью распознавания стандартных персептронов ансамбль спроектированных ИНС со встроенной процедурой отбора признаков позволяет повысить качество получаемых решений: относительное улучшение незначительно лишь для БД Emo-DB – 0.90%, в то время как для SAVEE, LEGO и UUDB оно составляет 3.41%, 4.27% и 4.05% соответственно. Можно также заключить, что в рамках данного эксперимента применение ансамбля полносвязных персептронов для каждой из рассматриваемых БД эффективнее в смысле значений метрики F-score по сравнению с результатами, полученными одним из сгенерированных персептронов.

Таблица 4.4 – Результаты тестирования подхода wrapper

Название БД	Классификатор	Значение F-score	Значение критерия K2	Число нейронов на скрытом слое	Число признаков
Emo-DB	Классический перцептрон (WEKA)	80.83	81759.8	197	384.0
	Перцептрон, спроектированный коллективным ГА, со встроенной процедурой отбора информативных признаков	79.94 (↓1.10%)	40997.0 (↑49.86%)	139.0	265.0
	Ансамбль перцептронов, спроектированных коллективным ГА, со встроенной процедурой отбора информативных признаков	81.56 (↑0.90%)	41895.7 (↑48.76%)	142.0	268.1
SAVEE	Классический перцептрон (WEKA)	59.55	81759.8	197	384.0
	Перцептрон, спроектированный коллективным ГА, со встроенной процедурой отбора информативных признаков	60.44 (↑1.49%)	46144.9 (↑43.56%)	155.7	269.2
	Ансамбль перцептронов, спроектированных коллективным ГА, со встроенной процедурой отбора информативных признаков	61.58 (↑3.41%)	43208.3 (↑47.15%)	149.6	261.5

Продолжение Таблицы 4.4

Название БД	Классификатор	Значение F-score	Значение критерия K2	Число нейронов на скрытом слое	Число признаков
LEGO	Классический перцептрон (WEKA)	68.19	80058.6	195	384.0
	Перцептрон, спроектированный коллективным ГА, со встроенной процедурой отбора информативных признаков	66.81 (↓2.02%)	44910.7 (↑43.90%)	151.7	273.0
	Ансамбль перцептронов, спроектированных коллективным ГА, со встроенной процедурой отбора информативных признаков	71.10 (↑4.27%)	44107.5 (↑44.91%)	149.9	271.7
UUDB	Классический перцептрон (WEKA)	49.34	80276.8	195	384.0
	Перцептрон, спроектированный коллективным ГА, со встроенной процедурой отбора информативных признаков	49.32 (↓0.04%)	45330.0 (↑43.53%)	154.0	272.0
	Ансамбль перцептронов, спроектированных коллективным ГА, со встроенной процедурой отбора информативных признаков	51.34 (↑4.05%)	43603.4 (↑45.68%)	148.8	269.2

Результаты распознавания эмоций человека по речи коллективом автоматически спроектированных ИНС со встроенной процедурой отбора признаков (таблица 4.4) сравнивались с результатами, полученными ранее в главе

3 (коллективом автоматически сгенерированных моделей, но без отбора атрибутов, таблица 3.5). Оказалось, что для всех рассматриваемых БД значения метрики F-score в рамках данных экспериментов статистически неразличимы (t-критерий Стьюдента при уровне значимости $p=0.05$). Для Emo-DB и SAVEE значение метрики F-score незначительно выше для случая без отбора признаков (82.90% против 81.56% и 62.02% против 61.58% соответственно), в то время как для LEGO и UUDB в ходе эксперимента с извлечением информативных атрибутов были получены чуть более лучшие значения метрики F-score (71.10% против 71.05% и 51.34% против 50.68% соответственно). Однако при сравнении значений критерия K2 «вычислительная сложность» для сгенерированных моделей процедура со встроенным отбором признаков становится более предпочтительной.

4.4 Сравнительный анализ подходов filter и wrapper на задаче распознавания эмоций человека по речи

Несмотря на то, что каждый из предложенных методов имеет свои преимущества, сравнительный анализ полученных результатов позволит выявить подход, рекомендуемый для задачи распознавания эмоций человека по речи. Безусловно, главным критерием является точность распознавания. Поэтому на основании значений метрики F-score выявим алгоритмическую схему, наиболее эффективную для каждой из рассматриваемых БД.

Для Emo-DB с помощью подхода filter в сочетании с коллективом классификаторов MLP, SVM и Logit удалось достичь значения метрики F-score = 86.26%. В то время как применение подхода wrapper обеспечило лишь 81.56%. Несмотря на существенное различие в абсолютных значениях метрики F-score, статистически данные результаты не различимы (t-критерий Стьюдента при уровне значимости $p=0.05$). Дополнительно можно отметить, что большего сокращения размерности входного вектора признаков удалось добиться с помощью подхода filter (166.9 против 268.1).

Применение схемы wrapper к SAVEE позволило достичь F-score = 61.58% при 261.5 извлеченных признаках. Схема filter обеспечила 68.71% при 165.9 атрибутах. И вновь в ходе статистической проверки значимых различий выявлено не было.

Для LEGO подходы wrapper и filter обеспечили примерно одинаковую точность распознавания: F-score = 71.10% и 71.29% соответственно при 271.7 и 150.7 отобранных атрибутах.

Лишь на UUDV схема wrapper незначительно превосходит filter на несколько десятых процента: F-score = 51.34% против 51.02% (число признаков 269.2 и 146.9 соответственно).

Однако для объективного сравнения нужно учитывать тот факт, что в сочетании с подходом filter было задействовано три различных вида классификаторов, в то время как в подходе wrapper использовались лишь полносвязные перцептроны. Поэтому был проведен дополнительный эксперимент, в котором эффективность разработанной схемы filter исследовалась в сочетании лишь с ИНС (машины опорных векторов и регрессия были исключены).

Как и ранее, островной многокритериальный ГА был применен для оптимизации системы критериев (4.7). При том же количестве вычислительных ресурсов и прежних настройках генетических операторов была найдена аппроксимация множества Парето, содержащая несравнимые между собой варианты сокращенных подсистем признаков, использующихся для обучения полносвязных перцептронов стандартной структуры. В качестве активационных функций использовались сигмоиды. За счет меньшего количества входных признаков было уменьшено и число нейронов на скрытом слое, определяемое по общепринятому правилу. В таблице 4.5 приведены результаты данного эксперимента.

Кроме значений метрики F-score и относительного улучшения, вычисляемого по сравнению с результатами, полученными на полном наборе

признаков стандартным перцептроном, приведены оценки вычислительной сложности нейросетей, обученных на сокращенных наборах признаков.

Таблица 4.5 – Подход filter в сочетании с полносвязными перцептронами

Название БД	Значение F-score, %	Вычислительная сложность ИНС
Emo-DB	84.89 (↑5.02%)	17507.2
SAVEE	68.83 (↑15.58%)	17321.2
LEGO	72.41 (↑6.19%)	13850.1
UUDB	50.27 (↑1.88%)	13411.45

Анализ результатов, представленных в таблице 4.5, показал, что для трех БД (Emo-DB, SAVEE, LEGO) значения F-score, полученные в ходе применения схемы filter, превосходят значения той же метрики, достигнутые с помощью схемы wrapper: разница составляет от 1.31 до 7.25%. И лишь на одной БД (UUDB) подход wrapper оказался более эффективным в смысле метрики F-score. Однако статистически значимых различий в сравниваемых значениях выявлено не было (t-критерий Стьюдента при уровне значимости $p=0.05$).

Интерес представляет второй критерий «вычислительная сложность», поскольку в результате применения подхода filter удалось получить более простые модели по сравнению с перцептронами, сгенерированными в рамках схемы wrapper. Данный факт объясняется тем, что в предложенной реализации схема wrapper направлена одновременно как на проектирование перцептронов, так и на отбор признаков, в связи с чем размерность задачи значительно возрастает. В свою очередь, подход filter представляет собой стадию предобработки данных и не включает этап построения моделей: на уже сокращенных БД обучаются перцептроны с более простыми структурами. А это значит, что для задачи распознавания эмоций человека по речи рекомендуется применение схемы filter,

которая может использоваться также и в сочетании с классификаторами различной природы. Стоит отметить, что представленный в данной работе подход *wrapper* будет более эффективным для БД меньшей размерности.

ВЫВОДЫ

Одним из способов повышения эффективности классификационных моделей является предобработка данных, в частности, отбор информативных признаков. В случае если стандартные методы снижения размерности не демонстрируют желаемого результата, предпочтение отдается эвристическим процедурам.

В данной главе для повышения качества решений предложены две схемы отбора релевантных атрибутов из баз данных. Первая *filter* основана на введении критериев, оценивающих статистические свойства обучающих примеров, а вторая *wrapper* оперирует функционалами, отражающими качество классификации на рассматриваемой подсистеме признаков. В обоих случаях была разработана двухкритериальная оптимизационная модель, а в качестве поискового алгоритма использовался островной ГА, представленный ранее.

Как было показано, применение разработанных схем к задаче распознавания эмоций человека по речи позволяет повысить качество получаемых решений. Однако в связи с высокой размерностью входного вектора признаков для данной задачи рекомендовано использовать схему *filter*, которая также может быть реализована и в сочетании с коллективом классификаторов различной природы.

Таким образом, разработанный коллективный многокритериальный ГА был успешно применен в области анализа речевых сигналов. При этом сочетание различных концепций эволюционного поиска позволило избежать выбора наиболее эффективного алгоритма, что является нетривиальной задачей даже для экспертов. Учитывая полученные результаты, можно утверждать, что цель диссертационного исследования достигнута полностью.

ЗАКЛЮЧЕНИЕ

В диссертационной работе получены следующие результаты:

1. Проведен анализ эффективности эволюционных алгоритмов многокритериальной оптимизации, основанных на разных концепциях. Было выявлено, что не существует метода, демонстрирующего наилучшие результаты для всех тестовых задач.

2. Разработан коллективный генетический алгоритм для решения задач многокритериальной оптимизации, отличающийся от своих компонент параллельной структурой, а также сочетанием различных концепций эволюционного поиска.

3. Разработан метод проектирования архитектуры нейросетевых классификаторов, позволяющий генерировать набор альтернативных моделей различной точности и вычислительной сложности.

4. Предложены эволюционные методы извлечения информативных признаков из баз данных при решении классификационных задач, основанные на двухкритериальных оптимизационных моделях.

5. Предложенные алгоритмические схемы были реализованы в виде программных систем и применены для решения задач распознавания эмоций человека по речи, что позволило повысить качество получаемых решений.

Таким образом, в диссертационной работе предложены, реализованы и проверены на тестовых и прикладных задачах новые алгоритмические схемы, основанные на эволюционных вычислениях, для решения задач многокритериальной оптимизации, предобработки данных и классификации. Представленные подходы обладают большей эффективностью по сравнению с известными аналогами, что является вкладом в теорию и практику обработки информации и интеллектуального анализа данных.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Айвазян, С.А., Бухштабер, В.М., Енюков, И.С., Мешалкин, Л.Д. Прикладная статистика. Классификация и снижение размерности. – М.: Финансы и статистика, 1989. – 607 с.
2. Архангельский, В.И., Богаенко, И.Н., Грабовский, Г.Г., Рюмшин, Н.А. Нейронные сети в системах автоматизации. – К.: Техника, 1999. – 234 с.
3. Васенков, Д.В. Методы обучения искусственных нейронных сетей. Компьютерные инструменты в образовании, 2007. – С. 20–29.
4. Воронцов, К.В. Лекции по методу опорных векторов [Электронный ресурс]. – Электрон. текстовые дан. – Режим доступа: <http://www.ccas.ru/voron/download/SVM.pdf> свободный.
5. Галушкин, А.И. Синтез многослойных систем распознавания образов. — М.: «Энергия», 1974.
6. Генетический алгоритм. Стандарт / Сост. В.В. Бухтояров, П.В. Галушин, А.Б. Сергиенко и др. [Электронный ресурс]. – Электрон. текстовые дан. – Режим доступа: http://www.harrix.org/files/61/Geneticheskii_algorithm_Standart_Part_I_v_1_8_Release_Candidate.pdf свободный.
7. Грибачев, В.П. Настоящее и будущее нейронных сетей. Компоненты и технологии, №5, 2006. – С. 28-32.
8. Грибачев, В.П. Элементная база аппаратных реализаций нейронных сетей. Компоненты и технологии, №8, 2006. – С. 72-75.
9. Заенцев, И. В. Нейронные сети: основные модели, Воронеж, 1999.
10. Каллан, Р. Основные концепции нейронных сетей. Пер. с англ. – М.: Издательский дом «Вильямс». 2001. – 288 с.
11. Короткий, С. Нейронные сети: алгоритм обратного распространения. [Электронный ресурс]. – Электрон. текстовые дан. – Режим доступа: http://lii.newmail.ru/kor_nn2.htm свободный.
12. Круглов, В.В., Борисов В. В. Искусственные нейронные сети теория и практика. – М.: Горячая линия – Телеком, 2002.

13. Круглов, В.В., Дли, М.И., Голунов, Р.Ю. Нечеткая логика и искусственные нейронные сети: Учеб. пособие. – М.: Издательство физико-математической литературы, 2001.
14. Лобанов, Б.М. Компьютерный синтез и клонирование речи / Б.М. Лобанов, Л.И. Цирульник // Минск: Белорусская Наука, 2008. – 344 с.
15. Минский, М., Пейперт, С. Перцептроны. – М.: Мир, 1971.
16. Розенблатт, Ф. Принципы нейродинамики: Перцептрон и теория механизмов мозга. – М.: Мир, 1965.
17. Рутковская, Д. Нейронные сети, генетические алгоритмы и нечеткие системы /Д. Рутковская, М. Пилиньский, Л. Рутковский; пер. с польск. И.Д. Рудинского. – М.: Горячая линия – Телеком, 2006. – 452 с.
18. Семенкин, Е.С., Ахмедова, Ш.А. Кооперативный бионический алгоритм безусловной оптимизации. Программные продукты и системы. 2013. № 4 (104). – С. 133–136.
19. Семенкина, М.Е. Самоадаптивные эволюционные алгоритмы проектирования информационных технологий интеллектуального анализа данных. Искусственный интеллект и принятие решений, 2013, №1. – С. 13–23.
20. Тынченко, В.С. Формирование распределенных систем структурно-параметрического синтеза нейросетевых моделей // Канд. техн. наук, 05.13.01. — СибГАУ, Красноярск, 2008, 165 с.
21. Яминов, Б. Генетические алгоритмы [Электронный ресурс] / Б. Яминов. – Электрон. текстовые дан. – Режим доступа: <http://rain.ifmo.ru/cat/view.php/theory/unsorted/genetic-2005> свободный.
22. Abraham, A., Jain, L., and Goldberg, R. Evolutionary multiobjective optimization: theoretical advances and applications. New York: Springer Science, 302 p., 2005.
23. Amato, F., López, A., Peña-Méndez, E.M., Vañhara, P., Hampl, A., Havel J. Artificial neural networks in medical diagnosis. Journal of Applied Biomedicine, Vol. 11, pp. 47–58, 2013.

24. Ang, J., Dhillon, R., Krupski, A., Shriberg, E., and Stolcke, A. Prosody-Based Automatic Detection of Annoyance and Frustration in Human-Computer Dialog, Proc. Eighth Int'l Conf. Spoken Language Processing (ICSLP), 2002.
25. Batliner, A., Fischer, K., Hubera, R., Spilker, J., and Noth, E. How to Find Trouble in Communication. *Speech Comm.*, vol. 40, pp. 117-143, 2003.
26. Bentley, P., Wakefield, J. Finding acceptable solutions in the pareto-optimal range using multiobjective genetic algorithms. *Soft Computing in Engineering Design and Manufacturing* 5, pp. 231-240, 1997.
27. Boersma, P. Praat, a system for doing phonetics by computer. *Glott international*, 5(9/10), pp. 341-345, 2002.
28. Breiman, L. Random Forests. *Machine Learning*, 45(1), pp. 5-32, 2001.
29. Breiman, L. Bagging predictors. *Machine Learning*, 24(2), pp. 123-140, 1996.
30. Brester, C., Semenkin, E. Development of adaptive genetic algorithms for neural network models multicriteria design. *Vestnik SibGAU*, vol. 4 (50), pp. 99-103, 2013.
31. Brester C., Semenkin E. Cooperative Multi-objective Genetic Algorithm with Parallel Implementation. *Advances in Swarm and Computational Intelligence*, LNCS 9140, pp. 471–478, 2015.
32. Brester, C.Yu., Semenkina, O.E., Sidorov, M.Yu. Robust and reliable techniques for speech-based emotion recognition. *Vestnik SibGAU*, vol. 16(1), pp. 28–34, 2015.
33. Brester, C., Semenkin, E., Sidorov, M. Speech-based emotion recognition: Application of collective decision making concepts. *International Conference on Computer Science and Artificial Intelligence (ICCSAI2014)*, pp. 216–220, 2014.
34. Brester, C., Semenkin, E., Sidorov, M., Kovalev, I., Zelenkov, P. Evolutionary feature selection for emotion recognition in multilingual speech analysis. *Proceedings of IEEE Congress on Evolutionary Computation (CEC2015)*, pp. 2406–2411, 2015.

35. Brester, C., Semenkin, E., Sidorov, M., Minker, W. Self-adaptive multi-objective genetic algorithms for feature selection. *Proceedings of International Conference on Engineering and Applied Sciences Optimization*, pp. 1838–1846, 2014.
36. Brockhoff, D., Zitzler, E. Improving hypervolume-based multiobjective evolutionary algorithms by using objective reduction methods. In: *Evolutionary Computation (CEC), 2007 IEEE Congress on*. IEEE, pp. 2086-2093, 2007.
37. Broomhead, D.S., Lowe, D. Multivariable functional interpolation and adaptive networks. *Complex Systems* 2, pp. 321–355, 1988.
38. Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier W. F., and Weiss, B. A database of german emotional speech. In *Interspeech*, pp. 1517–1520, 2005.
39. Coello Coello, Carlos A. A short tutorial on evolutionary multiobjective optimization. *Evolutionary Multi-Criterion Optimization*, pp. 21-40, 2001.
40. Coello Coello, Carlos A. Evolutionary multi-objective optimization: a historical view of the field. *Computational Intelligence Magazine, IEEE* 1 (1), pp. 28-36, 2006.
41. Coello Coello, Carlos A., Lamont, GB, Van Veldhuizen, DA. *Evolutionary algorithms for solving multi-objective problems*. Springer Science & Business Media, 2007.
42. Corne, D., Jerram, N., Knowles, J., Oates, M. Pesa-II: Region-based selection in evolutionary multiobjective optimization. In: *GECCO 2001: Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 283-290, 2001.
43. Corne, D., Knowles, J. Techniques for highly multiobjective optimisation: some nondominated points are better than others. In: *GECCO 2007: Proceedings of the Genetic and Evolutionary Computation Conference*, ACM, pp. 773-780, 2007.
44. Corne, D., Knowles, J., Oates, M. The pareto envelope-based selection algorithm for multiobjective optimization. In: *Parallel Problem Solving from Nature PPSN VI*. Springer, pp. 839-848, 2000.
45. Damasio, A.R. *Descartes' error: Emotion, reason, and the human brain*. Putnam's Sons, New York, NY, 1994.

46. Datta, R., Deb, K. A bi-objective based hybrid evolutionary-classical algorithm for handling equality constraints. In: *Evolutionary Multi-Criterion Optimization*. Springer, pp. 313-327, 2011.
47. Deb, K. *Multi-objective optimization using evolutionary algorithms*. Wiley, 2001.
48. Deb, K., Mohan, M., Mishra, S. Towards a Quick Computation of Well-Spread Pareto-Optimal Solutions. In: *Evolutionary Multi-Criterion Optimization*. Springer, pp. 222-236, 2003.
49. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6 (2), pp. 182-197, 2002.
50. Deb, K., Saxena, D. Searching for pareto-optimal solutions through dimensionality reduction for certain large-dimensional multi-objective optimization problems. In: *Proceedings of the World Congress on Computational Intelligence (WCCI-2006)*, pp. 3352-3360, 2006.
51. Ding, Sh., Li, H., Su, Ch., Yu, J., Jin, F. Evolutionary artificial neural networks: a review. *Artificial Intelligence Review*, vol. 39, pp. 251– 260, 2013.
52. Drechsler, N., Drechsler, R., Becker, B. Multi-objective optimisation based on relation favour. In: *Evolutionary Multi-Criterion Optimization*. Springer, pp. 154-166, 2001.
53. Edgeworth, F. Y. *Mathematical Psychics: An Essay on the Application of Mathematics to the Moral Sciences*. P. Keagan, London, England, 150 p., 1881.
54. Epps, J., Cowie, R., Narayanan, Sh., Schuller, B., Tao, J. Emotion and mental state recognition from speech. *EURASIP J. Adv. Sig. Proc.*, 2012.
55. Eyben, F., Wöllmer, M., Schuller, B. Opensmile: the munich versatile and fast opensource audio feature extractor. In *Proceedings of the international conference on Multimedia*, ACM, pp. 1459–1462, 2010.
56. Fonseca, C., Fleming, P. Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization. In: *Proceedings of the 5th*

International Conference on Genetic Algorithms. Morgan Kaufmann Publishers Inc., pp. 416–423, 1993.

57. Fonseca, C., Fleming, P. Multiobjective optimization and multiple constraint handling with evolutionary algorithms. I. A unified formulation. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans* 28 (1), pp. 26-37, 1998.

58. Fonseca, C., Fleming, P. On the performance assessment and comparison of stochastic multiobjective optimizers. In: *Parallel Problem Solving from Nature – PPSN VI*. Vol. 1141. Springer, pp. 584-593, 1996.

59. Freitas, A.A. *Data Mining and Knowledge Discovery with Evolutionary Algorithms*. Springer-Verlag, Berlin, 2002.

60. Friedman, J., Hastie, R., Tibshirani, R. *Additive Logistic Regression: a Statistical View of Boosting*. Stanford University, 1998.

61. Fürnkranz, J., Gamberger, D., Lavrač, N. *Foundations of Rule Learning*. Cognitive Technologies, 334 p. 2012.

62. Goldberg, D. *Genetic algorithms in search, optimization, and machine learning*. Addison-wesley, 1989.

63. Goutte, C., Gaussier, E. A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. *ECIR'05 Proceedings of the 27th European conference on Advances in Information Retrieval Research*, pp. 345–359, 2005.

64. Graciarena, M., Shriberg, E., Stolcke, A., Enos, F., Hirschberg, J., and Kajarekar, S. Combining Prosodic, Lexical and Cepstral Systems for Deceptive Speech Detection, *Proc. Int'l Conf. Acoustics, Speech and Signal Processing (ICASSP '06)*, vol. I, pp. 1033-1036, 2006.

65. Günther, F. and Fritsch, S. Neuralnet: Training of Neural Networks. *The R Journal*, vol. 2/1, pp. 30–38, 2010.

66. Hajela, P. and Lin, C.-Y. Genetic search strategies in multicriterion optimal design. *Structural Optimization*, 4, pp. 99–107, 1992.

67. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H. The WEKA Data Mining Software: An Update. SIGKDD Explorations, Volume 11, Issue 1, 2009.
68. Hamdani, T.M., Won, J., Alimi, A.M., Karray, F. Multi-objective feature selection with NSGA II. In Adaptive and Natural Computing Algorithms Lecture Notes in Computer Science, vol. 4431, pp. 240–247, 2007.
69. Хан, J. Data Mining. – [Электронный ресурс]. – Электрон. текстовые дан. – Режим доступа: <http://web.engr.illinois.edu/~hanj/pdf/ency99.pdf> свободный, 1999.
70. Наг, S., Jackson, P. Machine Audition: Principles, Algorithms and Systems, chapter Multimodal Emotion Recognition. IGI Global, Hershey PA, pp. 398–423, Aug. 2010.
71. Hirschberg, J., Benus, S., Brenier, J.M., Enos, F., and Friedman, S. Distinguishing Deceptive from Non-Deceptive Speech. Proc. Ninth European Conf. Speech Comm. and Technology (INTERSPEECH '05), pp. 1833-1836, 2005.
72. Holland, J.H. Adaptation in natural and artificial systems / J.H. Holland – Ann Arbor. MI: University of Michigan Press, 1975.
73. Holte, R.C. Very simple classification rules perform well on most commonly used datasets. Machine Learning, 11, pp.63-91, 1993
74. Horn, J., Nafpliotis, N., Goldberg, D. A niched pareto genetic algorithm for multiobjective optimization. In: Evolutionary Computation (CEC), 1994 IEEE Congress on. IEEE, pp. 82-87, 1994.
75. Huang, Y.A Comprehensive Study of Artificial Neural Networks. Advances in Artificial Neural Networks – Methodological Development and Application, vol. 2, pp. 973–1007, 2009.
76. Hughes, E. Evolutionary many-objective optimisation: many once or one many? In: Evolutionary Computation (CEC), 2005 IEEE Congress on. IEEE, pp. 222 - 227, 2005.
77. Inselberg, A. The plane with parallel coordinates. The Visual Computer 1 (2), pp. 69-91, 1985.

78. Ishibuchi, H. and Murata, T. Multi-objective genetic local search algorithm. In Proceedings of 1996 IEEE International Conference on Evolutionary Computation (ICEC'96), Piscataway, NJ, pp. 119–124, 1996.
79. Ishibuchi, H., Tsukamoto, N., Nojima, Y. Evolutionary many-objective optimization: A short review. In: Evolutionary Computation (CEC), 2008 IEEE Congress on. IEEE, pp. 2419-2426, 2008.
80. John, G.H., Enhancements to the data mining process. Ph.D. Thesis, Computer Science Department, Stanford University, CA, 1997.
81. John, G.H., Langley, P. Estimating Continuous Distributions in Bayesian Classifiers. In: Eleventh Conference on Uncertainty in Artificial Intelligence, San Mateo, pp. 338-345, 1995.
82. Keerthi, S.S., Shevade, S.K., Bhattacharyya, C., Murthy, K.R.K. Improvements to Platt's SMO Algorithm for SVM Classifier Design. Neural Computation, 13(3), pp. 637-649, 2001.
83. Khan, Z.H., Alin, T.Sh., and Hussain, Md.A. Price Prediction of Share Market using Artificial Neural Network (ANN). International Journal of Computer Applications, vol. 22, no. 2, pp. 42–47, 2011.
84. Kim, I., Weck, D.O. Adaptive weighted-sum method for bi-objective optimization: Pareto front generation. Structural and multidisciplinary optimization, 29 (2), pp. 149-158, 2005.
85. Kohavi R., John G.H. Wrappers for feature subset selection. Artificial Intelligence, 97, pp. 273-324, 1997.
86. Kovacs, T. Genetics-based Machine Learning. Handbook of Natural Computing: Theory, Experiments, and Applications. Springer Verlag, 2010, 62 p.
87. Kukkonen, S., Member, S., Lampinen, J. Ranking-Dominance and Many-Objective Optimization. In: Evolutionary Computation (CEC), 2007 IEEE Congress on. IEEE, pp. 3983-3990, 2007.
88. Kursawe, F. A variant of evolution strategies for vector optimization. In H.-P. Schwefel and R. Männer, editors, Parallel Problem Solving from Nature, pp. 193–197, 1991.

89. Kwon, O.W., Chan, K., Hao, J., and Lee, T.W. Emotion Recognition by Speech Signals, Proc. Eighth European Conf. Speech Comm. and Technology (EUROSPEECH), 2003.
90. Landwehr, N., Hall, M., Frank, E. Logistic Model Trees, 2005.
91. Lanzi, PL. Fast feature selection with genetic algorithms: a filter approach. IEEE International Conference on Evolutionary Computation, pp. 537–540, 1997.
92. Laumanns, M., Thiele, L., Deb, K., Zitzler, E. Combining convergence and diversity in evolutionary multiobjective optimization. *Evolutionary computation*, 10 (3), pp. 263-82, 2002.
93. Li, H., Zhang, Q. Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II. *IEEE Transactions on Evolutionary Computation* 13 (2), pp. 284-302, 2009.
94. Maneeratana, K., Boonlong, K., Chaiyaratana, N. Compressed-objective genetic algorithm. *Parallel Problem Solving from Nature. PPSN IX*, pp. 473-482, 2006.
95. Marchi, E., Schuller, B., Batliner A., Fridenzon Sh., Tal Sh., Golan O. Emotion in the speech of children with autism spectrum conditions: prosody and everything else. *WOCCI*, pp. 17-24, 2012.
96. Mitchell, T.M. *Machine learning*. McGraw-Hill, Maidenhead, U.K., International Student Edition, 414 p., 1997.
97. Mori, H., Satake, T., Nakamura, M., and Kasuya, H. Constructing a spoken dialogue corpus for studying paralinguistic information in expressive conversation and analyzing its statistical/acoustic characteristics. *Speech Communication*, 53, 2011.
98. Pareto, V. *Cours D'Economie Politique*, volume I and II. F. Rouge, Lausanne, 426 p., 1896.
99. Piatetsky-Shapiro, G., Frawley, W. *Knowledge Discovery in Databases*. AAAI/MIT Press, 1991.
100. Picard, R.W. *Affective computing*. MIT Press, Cambridge, MA, USA, 1997.

101. Picard, R.W. Affective computing. Tech. Rep. Perceptual Computing Section Technical Report No. 321, MIT Media Laboratory, 20 Ames St., Cambridge, MA 02139, 1995.
102. Platt, J. Fast Training of Support Vector Machines using Sequential Minimal Optimization. In B. Schoelkopf and C. Burges and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*, 1998.
103. Purshouse, R., Fleming, P. Evolutionary many-objective optimisation: an exploratory analysis. In: *Evolutionary Computation (CEC), 2003 IEEE Congress on*. Vol. 3. IEEE, pp. 2066-2073, 2003.
104. Purshouse, R., Fleming, P. On the evolutionary optimization of many conflicting objectives. *IEEE Transactions on Evolutionary Computation*, 11 (6), pp. 770-784, 2007.
105. Quinlan, R. Induction of decision trees. *Machine Learning*, 1(1), pp. 81-106, 1986.
106. Quinlan, R. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA, 1993.
107. Rumelhart, D.E., Hinton, G.E., Williams, R.J. Learning internal representations by error propagation. *Parallel Distributed Processing*, vol. 1, № 8, 1986.
108. Rune, A. Using Artificial Neural Networks To Forecast Financial Time Series. Master's thesis, Norwegian University of Science and Technology. Department of Computer and Information Science, 2010.
109. Sato, H., Aguirre, H., Tanaka, K. Controlling dominance area of solutions and its impact on the performance of MOEAs. In: *Evolutionary Multi-Criterion Optimization*. Springer, pp. 5-20, 2007.
110. Sato, H., Aguirre, H., Tanaka, K. Local dominance and local recombination in MOEAs on 0/1 multiobjective knapsack problems. *European Journal of Operational Research*, 181 (3), pp. 1708-1723, 2007.
111. Saxena, D.K., Duro, J.A., Tiwari, A., Deb, K., Zhang, Q. Objective reduction in many-objective optimization: Linear and nonlinear algorithms. *IEEE Transactions on Evolutionary Computation*, pp. 1-22, 2010.

112. Schaffer, D.J. Multiple Objective Optimization with Vector Evaluated Genetic Algorithms. In *Genetic Algorithms and their Applications: Proceedings of the First International Conference on Genetic Algorithms*, Hillsdale, NJ, pp. 93-100, 1985.

113. Schaffer, J.D. Multiple objective optimization with vector evaluated genetic algorithms. In J.J. Grefenstette, editor, *Proceedings of an International Conference on Genetic Algorithms and Their Applications*. Pittsburgh, PA, pp. 93–100, 1985.

114. Schaffer, D. Multiple objective optimization with vector evaluated genetic algorithms. In: *The 1st International Conference of Genetic Algorithms*. Pittsburgh, PA, pp. 93-100, 1985.

115. Schmitt, A., Ultes S., and Minker W. A parameterized and annotated corpus of the cmu let's go bus information system. *Proceedings of International Conference on Language Resources and Evaluation (LREC)*, 2012.

116. Schuller, B., Steidl, S., Batliner, A., Vinciarelli, A., Scherer R.K., Ringeval, F., Chetouani, M., Wenginger, F., Eyben, F., Marchi, E., Mortillaro, M., Salamin, H., Polychroniou, A., Valente, F., Kim, S. The INTERSPEECH 2013 computational paralinguistics challenge: social signals, conflict, emotion, autism. *INTERSPEECH 2013*, pp. 148-152, 2013.

117. Schuller, B., Steidl, S., Batliner, A.. The INTERSPEECH 2009 Emotion Challenge. *Tenth Annual Conference of the International Speech Communication Association*, 2009.

118. Silverman, B.W. *Density estimation for statistics and data analysis*. Chapman and Hall, London, 1986.

119. Squartini, S., Schuller, B., Hussain, A. Cognitive and Emotional Information Processing for Human-Machine Interaction. *Cognitive Computation*, 4(4), pp. 383-385, 2012.

120. Srinivas, N., Deb, K. Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary computation*, 2 (3), pp. 221 – 248, 1994.

121. Steidl, S., Levit, M., Batliner, A., Noth, E., and Niemann, H. Off All Things the Measure Is Man: Automatic Classification of Emotions and Inter-Labeler

Consistency. Proc. Int'l Conf. Acoustics, Speech, and Signal Processing (ICASSP '05), vol. 1, pp. 317-320, 2005.

122. Tanaka, M., Tanino, T. Global optimization by the genetic algorithm in a multiobjective decision support system. In: Proceedings of the Tenth International Conference on Multiple Criteria Decision Making, vol. 2, pp. 261-270, 1992.

123. Thrun, S.B. The Monk's problems: a performance comparison of different learning algorithms, Tech. Rept. CMU-CS-91-197, Carnegie Mellon University, Pittsburgh, PA, 1991.

124. Touchette, P., MacDonald, R., Langer, S. A scatter plot for identifying stimulus control of problem behavior. *Journal of Applied Behavior Analysis*, 18 (4), 343 p., 1985.

125. Väyrynen, E. Emotion recognition from speech using prosodic features. University of Oulu, Oulu, 2014.

126. Venkatadri, M., Srinivasa, R.K. A multiobjective genetic algorithm for feature selection in data mining. *International Journal of Computer Science and Information Technologies*, vol. 1, no. 5, pp. 443–448, 2010.

127. Volner, R. and Černák, I. Intelligent Communication Networks and Neural Network. *European International Journal of Science and Technology*, vol. 2, no. 6, pp. 29–40, 2013.

128. Wagner, T., Beume, N., Naujoks, B. Pareto-, aggregation-, and indicator-based methods in many-objective optimization. In: *Evolutionary Multi-Criterion Optimization*. Springer, pp. 742-756, 2007.

129. Wang, R. Preference-Inspired Co-evolutionary Algorithms. A thesis submitted in partial fulfillment for the degree of the Doctor of Philosophy, University of Sheffield, 231 p., 2013.

130. Whitley, D., Rana, S., and Heckendorn, R. Island model genetic algorithms and linearly separable problems. *Proceedings of AISB Workshop on Evolutionary Computation*, volume 1305 of LNCS, pp. 109-125, 1997.

131. Yang, J. and Hanovar, V. Feature subset selection using a genetic algorithm. *Journal of IEEE Intelligent Systems*, vol. 13, pp. 44-49, 1998.

132. Zaghoul, F.A., Al-Dhaheiri, S. Arabic Text Classification Based on Features Reduction Using Artificial Neural Networks. Proceedings of the conference on Computer Modelling and Simulation (UKSim), pp. 485 – 490, 2013.
133. Zhang, T., Hasegawa-Johnson, M., and Levinson, S.E. Children's Emotion Recognition in an Intelligent Tutoring Scenario. Proc. Eighth European Conf. Speech Comm. and Technology (INTERSPEECH), 2004.
134. Zhang, Q., Li, H. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. IEEE Transactions on Evolutionary Computation, 11 (6), pp. 712-731, 2007.
135. Zhang, Q., Liu, W., Li, H. The performance of a new version of moea/d on cec09 unconstrained mop test instances. In: Evolutionary Computation (CEC), 2009 IEEE Congress on. IEEE, pp. 203-208, 2009.
136. Zhang, Q., Zhou, A., Zhao, S., Suganthan, P. N., Liu, W., Tiwari, S. Multi-objective optimization test instances for the CEC 2009 special session and competition. University of Essex and Nanyang Technological University, Tech. Rep. CES-487, 2008.
137. Zhuo, L., Zheng, J., Wang, F., Li, X., Ai, B., Qian, J. A genetic algorithm based wrapper feature selection method for classification of hyperspectral images using support vector machine. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. XXXVII, part B7, pp. 397–402, 2008.
138. Zitzler, E., Laumanns, M., Bleuler, S. A Tutorial on Evolutionary Multiobjective Optimization. In: Gandibleux X., (eds.): Metaheuristics for Multiobjective Optimisation. Lecture Notes in Economics and Mathematical Systems, Vol. 535, Springer, 2004.
139. Zitzler, E., Laumanns, M., Thiele, L. SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization. Evolutionary Methods for Design Optimisation and Control with Application to Industrial Problems EUROGEN 2001, 3242 (103), pp. 95-100, 2002.
140. Zitzler, E., Thiele, L. Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach. IEEE Transactions on Evolutionary Computation, 3 (4), pp. 257 -271, 1999.

141. Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C., Fonseca, V. Performance assessment of multiobjective optimizers: an analysis and review. *IEEE Transactions on Evolutionary Computation*, 7 (2), pp. 117-132, 2003.

ПУБЛИКАЦИИ ПО ТЕМЕ ДИССЕРТАЦИИ

Статьи в ведущих рецензируемых научных журналах и изданиях

1. Брестер К.Ю., Бежитский С.С., Семенкина О.Э. Коллективный параллельный генетический алгоритм для решения задач многокритериальной оптимизации // *Системы управления и информационные технологии*, №4(62), 2015. – С. 34–38.

2. Brester C.Yu., Semenkina O.E., Sidorov M.Yu. Robust and reliable techniques for speech-based emotion recognition // *Вестник СибГАУ*. – Том 16. – № 1. – 2015. – С. 28–34.

3. Брестер К.Ю., Вишневская С.Р., Семенкина О.Э., Сидоров М.Ю. Эффективная процедура аутентификации студента по речи в дистанционном образовании // *Вестник СибГАУ*. – № 5(57). – 2014. – С. 51–57.

4. Брестер К.Ю., Вишневская С.Р., Семенкина О.Э. Распознавание психоэмоционального состояния дистанционного студента по устной речи адаптивными интеллектуальными информационными технологиями // *Вестник СибГАУ*. – №3 (55). – 2014. – С. 35–41.

5. Брестер К.Ю., Семенкин Е.С., Сидоров М.Ю. Система автоматического извлечения информативных признаков для распознавания эмоций человека в речевой коммуникации // *Программные продукты и системы*. – №4 (108). – 2014. – С. 127–131.

6. Brester C.Yu., Semenkin E.S. Development of adaptive genetic algorithms for neural network models multicriteria design // *Вестник СибГАУ*. – №4 (50). – 2013. – С. 99–103 с.

Публикации в изданиях, индексируемых в международных базах

7. Brester C., Semenkin E., Sidorov M., Kovalev I., Zelenkov P. Evolutionary feature selection for emotion recognition in multilingual speech analysis // Proceedings of the IEEE Congress on Evolutionary Computation (CEC2015), Sendai, Japan, 2015, pp. 2406–2411. (**Scopus**).
8. Brester C., Semenkin E., Sidorov M., Semenkina O. Multicriteria Neural Network Design in the Speech-Based Emotion Recognition Problem // Proceedings of the 12th International Conference on Informatics in Control, Automation and Robotics (ICINCO'2015), Colmar, France, 2015, vol. 1, pp. 621–628. (**Scopus**).
9. Brester C., Semenkin E. Cooperative Multi-Objective Genetic Algorithm with Parallel Implementation // Proceedings of the Sixth International Conference on Swarm Intelligence (ICSI 2015), Beijing, China, 2015, pp. 471–478. (**Scopus**).
10. Brester C., Semenkin E., Sidorov M., Minker W. Self-adaptive multi-objective genetic algorithms for feature selection // Proceedings of the International Conference on Engineering and Applied Sciences Optimization, Kos Island, Greece, 2014, pp. 1838–1846. (**Scopus**).
11. Brester C., Sidorov M., Semenkin E. Acoustic Emotion Recognition: Two Ways of Features Selection based on Self-Adaptive Multi-Objective Genetic Algorithm // Proceedings of the 11th International Conference on Informatics in Control, Automation and Robotics (ICINCO'2014), Vienna, Austria, 2014, vol. 2, pp. 851–855. (**Scopus**).
12. Brester C., Semenkin E., Sidorov M. Speech-based emotion recognition: Application of collective decision making concepts // Proceedings of the International Conference on Computer Science and Artificial Intelligence (ICCSAI2014), Wuhan, China, 2014, pp. 216–220. (**Web of Science**).
13. Sidorov M., Brester C., Semenkin E., Minker W. Speaker State Recognition with Neural Network-based Classification and Self-adaptive Heuristic Feature Selection // Proceedings of the 11th International Conference on Informatics in Control, Automation and Robotics (ICINCO'2014), Vienna, Austria, 2014, vol. 1, pp. 699–703. (**Scopus**).

14. Sidorov M., Brester C., Minker W., Semenkin E. Speech-Based Emotion Recognition: Feature Selection by Self-Adapted Multi-Criteria Genetic Algorithm // Proceedings of the 9th edition of the Language Resources and Evaluation Conference (LREC), Reykjavik, Iceland, 2014, pp. 3481–3485. (**Web of Science**).

Публикации в сборниках трудов конференций

15. Sidorov, M., Brester, Ch., Schmitt, A. Contemporary Stochastic Feature Selection Algorithms for Speech-based Emotion Recognition // Proc. of the Annual Conference of the International Speech Communication Association (INTERSPEECH'2015), Dresden, Germany, September 2015, p.192.

16. Брестер К.Ю., Бежитский С.С., Семенкин Е.С. Исследование эффективности механизмов самоадаптации генетического алгоритма в задачах многокритериальной оптимизации. // Материалы Третьей Всероссийской научной конференции с международным участием «Теория и практика системного анализа». – 2014. – С. 62–69.

17. Брестер К.Ю., Вишневская С.Р. Автоматическое извлечение информативных признаков в задаче распознавания состояния дистанционного студента // VIII международная конференция «Современные концепции научных исследований», Москва. – 2014. – С. 26–29.

18. Брестер К.Ю., Вишневская С.Р. Применение интеллектуальных информационных технологий для персонализации человеко-машинных коммуникаций в сфере дистанционного образования // Материалы Всероссийской научно-технической конференции ИТСиТ-2014, Кемерово, 2014. – С. 23–24.

19. Брестер К.Ю. Распознавание аэрокосмических снимков адаптивными интеллектуальными информационными технологиями // Тезисы докладов XX научно-технической конференции молодых ученых и специалистов, Королев. – 2014. – С. 654–655.

20. Брестер К.Ю., Бежитский С.С. Адаптивная модификация алгоритма МОЕА/D-DRA для решения задач бинарной оптимизации // Материалы XVIII Международной научной конференции «Решетневские чтения», Красноярск. – 2014. – Том 2. – С. 24–26.

21. Брестер К.Ю. Архитектура программной системы для решения задач классификации с автоматическим извлечением информативных признаков многокритериальным генетическим алгоритмом // Материалы XVIII Международной научной конференции «Решетневские чтения», Красноярск. – 2014. – Том 2. – С. 231–233.

22. Брестер К.Ю., Семенкин Е.С. Разработка параллельного адаптивного генетического алгоритма многокритериальной оптимизации // Пятая Международная конференция «Системный анализ и информационные технологии» САИТ-2013 (19-25 сентября 2013 г., г. Красноярск, Россия): Труды конференции. В 2-х т. – Т. 2. – Красноярск: ИВМ СО РАН, 2013. – С. 14-21.

23. Брестер К.Ю., Семенкин Е.С. Исследование эффективности алгоритмов кластеризации в методе SPEA // Интеллект и наука: труды XIII Междунар. науч. конф. – Железногорск, 2013. – С. 69-70.

24. Брестер К.Ю. Применение генетических алгоритмов для решения задач многокритериальной оптимизации // Всероссийский конкурс научно-исследовательских работ студентов и аспирантов в области информатики и информационных технологий: сб. науч. работ: в 3 т. – Белгород: ИД «Белгород», 2012. – Т. 3. – С. 355-356.

25. Брестер К.Ю. О самонастраивающемся генетическом алгоритме многокритериальной оптимизации // Материалы XVI Междунар. науч. конф., «Решетневские чтения». – Красноярск, 2012. – Ч. 2. – С. 474-475.

26. Брестер К.Ю. Применение адаптивных генетических алгоритмов для решения задач многокритериальной оптимизации // Сборник материалов Всероссийского конкурса научно-исследовательских работ студентов и аспирантов в области технических наук. – Санкт-Петербург: СПбГПУ, 2012. – С. 47–48.

Зарегистрированные программные системы

27. Брестер К.Ю., Сергиенко Р.Б. Программная система для реализации параллельного адаптивного генетического алгоритма многокритериальной оптимизации. – М.: Роспатент, 2013. № гос. рег. 2013618884.

28. Брестер К.Ю., Семенкин Е.С. Система автоматического генерирования нейросетевых моделей для решения задач классификации. – М.: Роспатент, 2013. № гос. рег. 2013660532.

29. Брестер К.Ю., Панфилов И.А., Семенкина О.Е. и др. Система автоматизированной классификации и категоризации мультилингвистических документов по содержанию. – М.: Роспатент, 2013. № гос. рег. 2013660992.

30. Брестер К.Ю., Семенкина О.Э. Коллективный эволюционный метод многокритериальной оптимизации на основе островной модели. – М.: Роспатент, 2015. № гос. рег. 2015662503.

31. Семенкин Е.С., Панфилов И.А., Брестер К.Ю. и др. Программная система автоматизированного проектирования интеллектуальных информационных сетей. – М.: Роспатент, 2015. № гос. рег. 2015662501.

32. Семенкин Е.С., Панфилов И.А., Брестер К.Ю. и др. Программная система для автоматизированной генерации моделей и алгоритмов решения задач анализа активности пользователя. – М.: Роспатент, 2015. № гос. рег. 2015662579.

Семенкин Е.С., Панфилов И.А., Брестер К.Ю. и др. Программная система для автоматизированного управления интеллектуальными информационными сетями. – М.: Роспатент, 2015. № гос. рег. 2015662581.

Приложение 1. Сравнение эффективности разработанного алгоритма с победителями соревнования CEC2009

Тестовая задача 1

ранг	Название алгоритма	IGD
1	MOEAD	0.00435
2	GDE3	0.00534
3	MOEADGM	0.0062
4	MTS	0.00646
5	LiuLiAlgorithm	0.00785
6	DMOEADD	0.01038
7	NSGAILS	0.01153
8	OWMOSaDE	0.0122
9	ClusteringMOEA	0.0299
10	AMGA	0.03588
11	MOEP	0.0596
	Кооперация NSGA-II, SPEA2, PICEA-g	0.06779
12	DECMOSA-SQP	0.07702
13	OMOEAI	0.08564

Тестовая задача 2

ранг	Название алгоритма	IGD
1	MTS	0.00615
2	MOEADGM	0.0064
3	DMOEADD	0.00679
4	MOEAD	0.00679
5	OWMOSaDE	0.0081
6	GDE3	0.01195
7	LiuLiAlgorithm	0.0123
8	NSGAILS	0.01237
9	AMGA	0.01623
10	MOEP	0.0189
11	ClusteringMOEA	0.0228
12	DECMOSA-SQP	0.02834
13	OMOEAI	0.03057
	Кооперация NSGA-II, SPEA2, PICEA-g	0.055517

Тестовая задача 3

ранг	Название алгоритма	IGD
1	MOEAD	0.00742
2	LiuLiAlgorithm	0.01497
3	DMOEADD	0.03337
4	MOEADGM	0.049
5	MTS	0.0531
6	ClusteringMOEA	0.0549
7	AMGA	0.06998
8	DECMOSA-SQP	0.0935
9	MOEP	0.099
10	OWMOSaDE	0.103
11	NSGAILS	0.10603
12	GDE3	0.10639
	Кооперация NSGA-II, SPEA2, PICEA-g	0.20214
13	OMOEAI	0.27141

Тестовая задача 4

ранг	Название алгоритма	IGD
1	MTS	0.02356
2	GDE3	0.0265
3	DECMOSA-SQP	0.03392
4	AMGA	0.04062
5	DMOEADD	0.04268
6	MOEP	0.0427
7	LiuLiAlgorithm	0.0435
8	OMOEAI	0.04624
9	MOEADGM	0.0476
10	OWMOSaDE	0.0513
	Кооперация NSGA-II, SPEA2, PICEA-g	0.05836
11	NSGAILS	0.0584
12	ClusteringMOEA	0.0585
13	MOEAD	0.06385

Тестовая задача 5

ранг	Название алгоритма	IGD
1	MTS	0.01489
2	GDE3	0.03928
3	AMGA	0.09405
4	LiuLiAlgorithm	0.16186
5	DECMOSA-SQP	0.16713
6	ОМОЕАII	0.1692
7	МОЕАD	0.18071
8	МОЕP	0.2245
9	ClusteringMOEA	0.2473
10	DМОЕАD	0.31454
	Кооперация NSGA-II, SPEA2, PICEA-g	0.33777
11	ОWМОСаDE	0.4303
12	NSGAILLS	0.5657
13	МОЕАDGM	1.7919

Тестовая задача 6

ранг	Название алгоритма	IGD
1	МОЕАD	0.00587
2	MTS	0.05917
3	DМОЕАD	0.06673
4	ОМОЕАII	0.07338
5	ClusteringMOEA	0.0871
6	МОЕP	0.1031
7	DECMOSA-SQP	0.12604
8	AMGA	0.12942
9	LiuLiAlgorithm	0.17555
10	ОWМОСаDE	0.1918
11	GDE3	0.25091
	Кооперация NSGA-II, SPEA2, PICEA-g	0.25360
12	NSGAILLS	0.31032
13	МОЕАDGM	0.5563

Тестовая задача 7

ранг	Название алгоритма	IGD
1	MOEAD	0.00444
2	LiuLiAlgorithm	0.0073
3	MOEADGM	0.0076
4	DMOEADD	0.01032
5	MOEP	0.0197
6	NSGAIILS	0.02132
7	ClusteringMOEA	0.0223
8	DECMOSA-SQP	0.02416
9	GDE3	0.02522
10	OMOEAI	0.03354
11	MTS	0.04079
12	AMGA	0.05707
13	OWMOSaDE	0.0585
	Кооперация NSGA-II, SPEA2, PICEA-g	0.08408

Тестовая задача 8

ранг	Название алгоритма	IGD
1	MOEAD	0.0584
2	DMOEADD	0.06841
3	LiuLiAlgorithm	0.08235
4	NSGAIILS	0.0863
5	OWMOSaDE	0.0945
6	MTS	0.11251
7	AMGA	0.17125
8	OMOEAI	0.192
9	DECMOSA-SQP	0.21583
10	ClusteringMOEA	0.2383
11	MOEADGM	0.2446
12	GDE3	0.24855
	Кооперация NSGA-II, SPEA2, PICEA-g	0.25885
13	MOEP	0.423

Тестовая задача 9

ранг	Название алгоритма	IGD
1	DMOEADD	0.04896
2	NSGAIILS	0.0719
3	MOEAD	0.07896
4	GDE3	0.08248
5	LiuLiAlgorithm	0.09391
6	OWMOSaDE	0.0983
7	MTS	0.11442
8	DECMOSA-SQP	0.14127
9	MOEADGM	0.1878
10	AMGA	0.18861
11	OMOEAI	0.23179
12	ClusteringMOEA	0.2934
	Кооперация NSGA-II, SPEA2, PICEA-g	0.31363
13	MOEP	0.342

Тестовая задача 10

ранг	Название алгоритма	IGD
1	MTS	0.15306
2	DMOEADD	0.32211
3	AMGA	0.32418
4	MOEP	0.3621
5	DECMOSA-SQP	0.36985
6	ClusteringMOEA	0.4111
7	GDE3	0.43326
8	LiuLiAlgorithm	0.44691
9	MOEAD	0.47415
	Кооперация NSGA-II, SPEA2, PICEA-g	0.53350
10	MOEADGM	0.5646
11	OMOEAI	0.62754
12	OWMOSaDE	0.743
13	NSGAIILS	0.84468