

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа ИЯТШ

Направление подготовки: 01.04.02 «Прикладная математика и информатика»

Отделение экспериментальной физики

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

Тема работы
Разработка программного продукта для обработки данных рентгеновских спектров

УДК 004.415.2:539.26

Студент

Группа	ФИО	Подпись	Дата
0ВМ92	Петрович Евгений Витальевич		

Руководитель

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОЭФ	Семёнов М.Е.	к.ф.-м.н., доцент		

Со-руководитель (по разделу «Концепция стартап-проекта»)

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ШИП	Ермушко Ж.А.	к.э.н., доцент		

КОНСУЛЬТАНТЫ:

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ООД	Антоневич О.А.	к.б.н., доцент		

ДОПУСТИТЬ К ЗАЩИТЕ:

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОЭФ	Семёнов М.Е.	к.ф.-м.н., доцент		

Томск – 2021

Планируемые результаты обучения по ООП

Код результата	Результат обучения
ПК(У)-1	Способен проводить научные исследования и получать новые научные и прикладные результаты самостоятельно и в составе научного коллектива
ПК(У)-2	Способен проводить поиск и анализ научной и научно-технической литературы по тематике проводимых исследований
ПК(У)-3	Способен разрабатывать и анализировать показатели качества информационных систем, используемых в производственной деятельности
ПК(У)-4	Способен планировать научно-исследовательскую деятельность, анализировать риски, управлять проектами, управлять командой проекта
ПК(У)-5	Способен к преподаванию математических дисциплин и информатики в образовательных организациях высшего образования
ПК(У)-6	Способен проектировать и организовывать учебный процесс по образовательным программам с использованием современных образовательных технологий
ОПК(У)-1	Способен решать актуальные задачи фундаментальной и прикладной математики
ОПК(У)-2	Способен совершенствовать и реализовывать новые математические методы решения прикладных задач
ОПК(У)-3	Способен разрабатывать математические модели и проводить их анализ при решении задач в области профессиональной деятельности
ОПК(У)-4	Способен комбинировать и адаптировать существующие информационно-коммуникационные технологии для решения задач в области профессиональной деятельности с учетом требований информационной безопасности
УК(У)-1	Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, вырабатывать стратегию действий
УК(У)-2	Способен управлять проектом на всех этапах его жизненного цикла
УК(У)-3	Способен организовывать и руководить работой команды, вырабатывая командную стратегию для достижения поставленной цели
УК(У)-4	Способен применять современные коммуникативные технологии, в том числе на иностранном (-ых) языке(-ах), для академического и профессионального взаимодействия
УК(У)-5	Способен анализировать и учитывать разнообразие культур в процессе межкультурного взаимодействия
УК(У)-6	Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение
высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Инженерная школа ядерных технологий
Направление подготовки 01.04.02 «Прикладная математика и информатика»
Отделение экспериментальной физики

УТВЕРЖДАЮ:

Руководитель ООП

_____ Семенов М.Е.

(Подпись) (Дата) (Ф.И.О.)

**ЗАДАНИЕ
на выполнение выпускной квалификационной работы**

В форме:

Магистерской диссертации

Студенту:

Группа	ФИО
ОВМ92	Петровичу Евгению Витальевичу

Тема работы:

Разработка программного продукта для обработки данных рентгеновских спектров	
Утверждена приказом директора (дата, номер)	№74-2/с от 15.03.2021

Срок сдачи студентом выполненной работы:	31.05.2021
--	------------

ТЕХНИЧЕСКОЕ ЗАДАНИЕ:

Исходные данные к работе	<i>Регламент по организации и проведению государственной итоговой аттестации по образовательным программам высшего образования - программам бакалавриата, программам специалитета и программам магистратуры в ТПУ, рентгенофлуоресцентные спектры полученные спектрометром CPB-1B</i>
---------------------------------	---

Перечень подлежащих исследованию, проектированию и разработке вопросов	<ol style="list-style-type: none"> 1. Провести сравнительный анализ литературы по теме диссертации. 2. Разработать алгоритм работы и архитектуру программы для обработки спектров. 3. Разработать техническую документацию. 4. Реализация программного продукта на языке программирования. 5. Подготовка и проведение тестирования программы.
Перечень графического материала	<ol style="list-style-type: none"> 1. Диаграммы алгоритмов работы программного продукта. 2. Диаграммы архитектуры программного продукта. 3. Скриншоты работы программного продукта.
Консультанты по разделам выпускной квалификационной работы	
<i>(если необходимо, с указанием разделов)</i>	
Раздел	Консультант
Финансовый менеджмент, ресурс эффективность и ресурсосбережение	Доцент ШИП, к.э.н. Ермушко Жанна Александровна.
Социальная ответственность	Доцент ООД, к.э.н. Антоневиц Ольга Алексеевна.
Иностранный язык	Старший преподаватель ОИЯ Утятина Янина Викторовна

Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику	15.03.2021
---	------------

Задание выдал руководитель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОЭФ	Семенов Михаил Евгеньевич	к.ф.-м. н., доцент		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
0ВМ92	Петрович Евгений Витальевич		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «КОНЦЕПЦИЯ СТАРТАП-ПРОЕКТА»

Студенту:

Группа	ФИО
0ВМ92	Петровичу Евгению Витальевичу

Школа	ИЯТШ	Направление	01.04.02 «Прикладная математика и информатика»
Уровень образования	Магистратура		

Перечень вопросов, подлежащих разработке:

<i>Проблема конечного потребителя, которую решает продукт, который создается в результате выполнения НИОКР (функциональное назначение, основные потребительские качества)</i>	
<i>Способы защиты интеллектуальной собственности</i>	
<i>Объем и емкость рынка</i>	
<i>Современное состояние и перспективы отрасли, к которой принадлежит представленный в ВКР продукт</i>	
<i>Себестоимость продукта</i>	
<i>Конкурентные преимущества создаваемого продукта</i>	
<i>Сравнение технико-экономических характеристик продукта с отечественными и мировыми аналогами</i>	
<i>Целевые сегменты потребителей создаваемого продукта</i>	
<i>Бизнес-модель проекта</i>	
<i>Производственный план</i>	
<i>План продаж</i>	

Перечень графического материала:

<i>При необходимости представить эскизные графические материалы (например, бизнес-модель)</i>	
---	--

Дата выдачи задания для раздела по линейному графику

Задание выдал консультант по разделу «Концепция стартап-проекта» (со-руководитель ВКР):

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ШИП	Ермушко Ж.А.	к.э.н., доцент		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
0ВМ92	Петрович Евгений Витальевич		

ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
0ВМ92	Петрович Евгений Витальевич

Школа	ИЯТШ	Отделение (НОЦ)	Отделение экспериментальной физики
Уровень образования	Магистратура	Направление/специальность	01.04.02 «Прикладная математика и информатика»

Тема ВКР:

Разработка программного продукта для обработки данных рентгеновских спектров	
Исходные данные к разделу «Социальная ответственность»:	
<p>1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения</p>	<p>Объект исследования: программный продукт для обработки рентгеновских спектров. Рабочее место: лаборатории промышленных предприятий. Оборудование: рентгенофлуоресцентный спектрометр, персональный компьютер. Область применения: количественный и качественный элементный анализ технологических веществ и материалов.</p>
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
<p>1. Правовые и организационные вопросы обеспечения безопасности:</p> <ul style="list-style-type: none"> – специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства; – организационные мероприятия при компоновке рабочей зоны. 	<p>1. СП 2.4.3648-20 "Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи"</p> <p>2. СанПиН 2.6.1.2523-09 "Нормы радиационной безопасности (НРБ-99/2009)"</p>
<p>2. Производственная безопасность:</p> <p>2.1. Анализ выявленных вредных и опасных факторов</p> <p>2.2. Обоснование мероприятий по снижению воздействия</p>	<p>Опасные и вредные производственные факторы:</p> <p>1. опасные и вредные производственные факторы, связанные со световой средой;</p> <p>2. опасные и вредные производственные факторы, связанные с аномальными микроклиматическими параметрами воздушной среды на местонахождении работающего</p>

	<p>3. коротковолновое электромагнитное излучение (поток фотонов высоких энергий) - рентгеновское излучение и гамма-излучение;</p> <p>4. повышенный уровень и другие неблагоприятные характеристики шума;</p> <p>5. производственные факторы, связанные с электрическим током, вызываемым разницей электрических потенциалов, под действие которого попадает работающий.</p>
3. Экологическая безопасность:	Атмосфера, Гидросфера, Литосфера: Воздействие на окружающую среду при утилизации компьютерной техники.
4. Безопасность в чрезвычайных ситуациях:	<p>Возможные ЧС в рабочей зоне: техногенные - производственные аварии и пожары.</p> <p>Наиболее типичной ЧС являются пожары на производстве.</p>

Дата выдачи задания для раздела по линейному графику	
---	--

Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ООД	Антоневич Ольга Алексеевна	к.б.н., доцент		

Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
0ВМ92	Петрович Евгений Витальевич		

Реферат

Выпускная квалификационная работа включает в себя: 133 страницы, 17 таблиц, 36 рисунков, 40 библиотечных ссылок, 21 формула, 3 приложения.

Место проведения работ - ТОО «ТехноАналит», Республика Казахстан, г. Усть-Каменогорск, ул. Крылова 85.

Ключевые слова: энергодисперсионный спектрометр, рентгенофлуоресцентный спектр, градуировка, химическая концентрация, нейронная сеть.

Предмет исследования – проблемы обработки данных рентгеновских спектров для определения химических концентраций веществ.

Цель работы – разработка программного продукта для создания качественных градуировочных моделей для определения химических концентраций веществ с помощью рентгенофлуоресцентных энергодисперсионных спектрометров.

Исследовательские и прикладные работы проводились с использованием рентгенофлуоресцентного энергодисперсионного спектрометра СРВ-1В и следующих инструментальных средств:

- Язык программирования C++,
- Кроссплатформенный фреймворк Qt 5.15.0,
- IDE Qt Creator,
- Интегрированный в Qt Creator редактор UML моделей,
- Язык программирования Python 3.6
- Doxygen – инструментальное средство для автоматической генерации документации проекта,
- Система контроля версий Git.

В работе изучены возможности повышения качества создаваемых градуировочных моделей, повышения уровня автоматизации создания градуировочных моделей, разработаны и предложены меры, позволяющие использовать программный продукт для создания градуировочных моделей для

спектрометра персоналом не обладающем высокой квалификацией в области спектрального анализа.

Практическая ценность работы – упрощение работ по определению химических концентраций веществ с помощью спектрометров, расширение круга персонала обладающих возможностью работы со спектрометрами.

Степень внедрения: средняя – проходит тестовые испытания.

Область применения: лаборатории химического анализа предприятий различного профиля.

Экономическая эффективность/значимость работы: высокая – значительное снижение затрат на эксплуатацию спектрометра.

Содержание

Введение.....	13
1. Обзор литературы	17
2. Теоретическая основа исследования и постановка задачи.....	18
2.1 Градуировка спектрометра.....	18
2.2 Теоретическая основа нейронной сети	24
2.2.1 Структура нейронной сети.....	24
2.2.2 Обучение нейронной сети	27
2.2.3 Регуляризация параметров нейронной сети.....	28
2. Разработка программного продукта.....	30
2.1 Решение проблемы выбора признаков.....	30
2.2 Решение проблемы малой выборки	34
2.3 Выбор необходимых инструментальных средств	35
3. Реализация программного продукта	36
3.1 Тестовая версия программного продукта.....	36
3.2 Архитектура нейронной сети.....	38
3.3. Интеграция программного продукта в программное обеспечение спектрометра СРВ-1В.....	42
3.4 Реализация интегрального программного модуля.	48
4. Тестирование и результаты.....	51
Заключение	57
5. Концепция стартап-проекта	58
5.1 Проблема конечного потребителя.....	58
5.2 Способы защиты интеллектуальной собственности	60
5.3 Объем и емкость рынка	60

5.4 Современное состояние и перспективы отрасли	61
5.5 Конкурентные преимущества создаваемого продукта	62
5.6 Целевые сегменты потребителей создаваемого продукта.....	63
5.6.1 Модель B2B	63
5.6.2 Модель B2C	63
5.7 Цена программного продукта	64
5.8 Финансирование и продвижение программного продукта	64
5.9 Бизнес-модель проекта	65
5.9.1 Команда проекта	65
5.9.2 Первоначальные инвестиции	66
5.9.3 Амортизация оборудования	68
5.9.4 Годовые затраты проекта	68
5.9.5 Разработка программного продукта.....	69
5.9.6 Себестоимость разработки программного продукта.....	69
5.10 Оценка эффективности.....	70
5.11 Заключение	74
6. Социальная ответственность	75
6.1 Правовые и организационные вопросы обеспечения безопасности	76
6.1.1 Специальные (характерные для проектируемой рабочей зоны) правовые нормы трудового законодательства	76
6.1.2 Основные эргономические требования к компоновке рабочей зоны.	77
6.2 Производственная безопасность	80
6.2.1 Анализ опасных и вредных производственных факторов.....	82
6.2.2 Обоснование мероприятий по снижению уровней воздействия опасных и вредных факторов на исследователя (работающего)	89

6.3 Экологическая безопасность.....	90
6.4 Безопасность в чрезвычайных ситуациях.....	91
6.4.1 Анализ возможных чрезвычайных ситуаций (ЧС).....	91
6.4.2 Меры по предупреждению возникновения пожара	91
6.4.3 Действия в случае возникновения пожара	92
6.5 Заключение	93
Список использованных источников	94
Список публикаций.....	97
Приложение А	98
Приложение Б	120
Приложение В.....	132

Введение

В настоящее время для качественного и количественного анализа химического состава веществ в промышленности широко применяются приборы, принцип работы которых основан на получении и анализе характеристических рентгеновских спектров этих веществ. В основу метода положен закон Мозли, связывающий энергию спектральных линий характеристического рентгеновского излучения (ХРИ) атома химического элемента с его порядковым номером. Преимущества химического анализа по характеристическому рентгеновскому излучению по сравнению с другими методами анализа заключаются в следующем:

- относительная простота;
- относительно невысокая стоимость;
- неразрушающий характер анализа;
- возможен анализ в любом агрегатном состоянии вещества;
- невысокие требования к подготовке образцов;
- возможность проведения одновременного анализа для нескольких элементов;
- быстрота проведения анализа.

Одним из разновидностей инструментальных средств анализа по спектрам ХРИ являются рентгеновские энергодисперсионные спектрометры. Их основным преимуществом перед масс-спектрометрами, спектрометрами с волновой дисперсией и другими подобными инструментальными средствами является относительная простота конструкции, надежность и невысокая стоимость самих приборов и их обслуживания.

Процесс количественного элементного спектрального анализа состоит из двух основных частей:

- получения и сохранения рентгеновского спектра ХРИ исследуемого материала при помощи аппаратной части спектрометра;
- обработка рентгеновского спектра ХРИ на компьютере при помощи специального программного обеспечения для получения информации о химическом составе исследуемого образца.

Для обработки рентгеновских спектров и определения значений концентраций химических элементов в исследуемом материале существует несколько различных методов. Все они требуют первичной предобработки спектров включающей в себя следующие шаги:

- подавление шума,
- удаление фона,
- выделение пиков линий ХРИ присутствующих элементов и определение их интенсивностей.

Далее полученные таким образом данные используются для определения количественного элементного состава исследуемого материала при помощи одного из следующих методов:

- метода фундаментальных параметров (МФП);
- методами основанными на стандартных образцах, как правило, полиномиальной регрессии.

Эти классические принципы и методы спектрального анализа подробно описаны в [1-6]

Таким образом процесс количественного элементного анализа спектра представляет собой сложный поэтапный процесс. Алгоритм этого процесса содержится в градуировке - программном объекте, в котором содержатся:

- значения настроек спектрометра, при которых должен быть снят спектр (экспозиция в секундах, коэффициент усиления детектора и т.д.),

- алгоритм первичной предобработки и математическая модель - градуировочное уравнение, реализующее метод, по которому определяются концентрации.

Для создания градуировок требуется привлечение опытных высококвалифицированных специалистов, работа которых заключается в ручном подборе всех параметров градуировки. Как правило, привлекаются специалисты непосредственно фирмы-производителя спектрометра либо их авторизованные представители, что является фактором значительно осложняющим и удорожающим применение спектрометров.

В настоящее время активно ведутся исследования применения технологий глубокого обучения нейронных сетей для анализа спектров для различных прикладных задач [7-16].

Нейронные сети хорошо зарекомендовали себя в решении самых различных задач регрессии и классификации.

Тем не менее, анализ продукции ведущих мировых фирм-производителей спектрометров, таких как «Shimadzu» [17], «Bruker» [18], «PANalytical» [19], «ThermoFisher Scientific» [20], «Oxford Instruments Industrial Analysis» [21] и др., показал, что на настоящий момент в их программном обеспечении используются только классические аналитические методы.

Хотя нейронные сети глубокого обучения давно используются для решения задач регрессии, применение нейронной сети глубокого обучения для создания градуировки для спектрометра имеет проблему выбора признаков [7].

Если на вход нейронной сети в качестве признаков подавать значения интенсивностей в выделенных пиках спектра, это мало отличается от классического методов спектрального анализа. Специалист должен выделить признаки из спектров, выбрать важные признаки, определить необходимость нормировки, проверить все признаки на мультиколлинеарность и т.д.

Если на вход нейронной сети подавать значения интенсивностей в каналах спектра, то в этом случае признаки будут обладать сильной мультиколлинеарностью, так как интенсивность каждого пика характеристического излучения распределена по нескольким каналам. В этом случае регрессионная модель будет обладать высокой статистической неопределенностью, то есть полученные при помощи такой модели оценки будут обладать неприемлемой дисперсией.

Целью настоящей магистерской диссертации является разработка, реализация в виде программного продукта и исследование метода для создания градуировок, основанного на использовании алгоритмов глубокого обучения нейросетей.

Научная и практическая новизна и значимость работы. В предложенном методе решена проблема мультиколлинеарности интенсивностей в каналах спектра. Применение метода не требует привлечения квалифицированных специалистов, что влечёт значительное снижение затрат на эксплуатацию спектрометра, и позволяет создавать качественные градуировки силами персонала лаборатории. Также, благодаря возможности создавать сложные градуировки, возможно расширение областей использования спектрального анализа.

Результаты работы. Разработанный программный продукт и созданные с его помощью градуировки были представлены специалистам ТОО «ТехноАналит» (Казахстан), специалистам уранодобывающих предприятий и предприятий цветной металлургии. Работа получила высокие оценки и в настоящее время программный продукт проходит тестовые испытания в химических лабораториях горнодобывающих предприятий Казахстана.

1. Обзор литературы

Последнее время технологии с применением нейронных сетей позволили достичь значительных результатов в различных областях человеческой деятельности. При помощи нейронных сетей решены многие проблемы, такие как машинное зрение и распознавание образов, распознавание естественного языка, с использованием нейронных сетей создаются автономные автомобили и роботы.

Также активные работы идут в области применения алгоритмов нейронных сетей для различных задач обработки спектральной информации. В [7] рассмотрены вопросы применения нейронных сетей в спектрометрии радиоактивных излучений. В работе продемонстрировано использование полносвязной нейронной сети для оценки качества спектров и определения наличия источников радиоактивного излучения. Тем не менее авторы отмечают проблемы при работе «с целым спектром, а не с отдельными пиками излучения». Применение технологии нейронных сетей в задаче обнаружения взрывчатых веществ рассмотрена в [22]. В работе также рассматривалась полносвязная нейронная сеть для задач идентификации вещества. В [13] авторы применяли полносвязные и сверточные нейронные сети для определения позиций пиков на спектре. Технологии нейронных сетей в количественном спектральном анализе рассмотрены в [15]. Авторы использовали 1400 стандартных образцов 23 материалов и свёрточные нейронные сети для определения концентраций относительно легких элементов. В [8] опубликована работа в которой при помощи нейронных сетей определялась концентрация золота. Так как исходный набор данных был относительно небольшой, в работе был сгенерирован искусственный массив данных спектров методом Монте-Карло.

В работах отмечаются трудности, связанные с применением нейронных сетей в спектральном анализе. К ним относятся ограниченные наборы научных данных, сбор которых требует больших затрат, тогда как для целей рекламы или распознавания образов можно получить большое количество дешевых данных.

Кроме того, работа с такими сырыми данными, как значения интенсивностей в каналах спектра имеют ряд проблем.

В последние годы опубликовано несколько моделей нейронных сетей применительно к спектральному анализу. В большинстве работ не рассматривались задачи связанные с количественным анализом. Если такой анализ и проводился, то данные в спектре подвергались предобработке для извлечения из них более абстрактных признаков.

Анализ промышленных разработок позволяет заключить, что проблемы, связанные с использованием нейронных сетей в количественном спектральном анализе, не решены. Все фирмы-производители спектрометрического оборудования продолжают использовать в своих продуктах традиционные методы анализа, требующие затрат временных, интеллектуальных и финансовых ресурсов.

Тем не менее, растущее количество опубликованных научных работ и некоторый прогресс в исследовании и применении нейронных сетей для количественного спектрального анализа позволяет заключить, что интерес к теме имеется и предпринимаются усилия для нахождения способов и методов преодоления связанных с этим трудностей.

2. Теоретическая основа исследования и постановка задачи

2.1 Градуировка спектрометра

Каждый химический элемент при воздействии на него возбуждающего рентгеновского излучения испускает характеристическое излучение с уникальным набором спектральных линий. Анализируемый спектр представляет собой сложную суперпозицию линий характеристического излучения от всех химических элементов, содержащихся в исследуемом материале, рассеянного излучения рентгеновской трубки и других артефактов. Интенсивность характеристического излучения в соответствующих линиях спектра зависит от

концентрации химических элементов в исследуемом материале. Кроме того, химические элементы в исследуемом материале оказывают взаимное влияние на характеристическое излучение друг друга ослабляя или, наоборот, усиливая его.

Интенсивность излучения выражается в количестве рентгеновских фотонов, испущенных материалом, подвергнутым рентгеновскому облучению, и представляет собой функциональную зависимость:

$$I = \int_{t_0}^{t_0+\tau} N(e, t) dt, \quad (1)$$

где $I \in \mathbb{N}$ – интенсивность излучения, $N \in \mathbb{N}$ – скорость счета фотонов за единицу времени, $e \in \mathbb{R}$ – энергия фотонов, $t \in \mathbb{R}$ – время, $\tau \in \mathbb{R}$ – время экспозиции.

В течение времени экспозиции спектрометром фиксируются фотоны излучения от исследуемого материала и их количество распределяется в соответствии с их энергиями по дискретизированной энергетической шкале. Таким образом, измеренный спектр можно представить в виде интервального вариационного ряда с равноинтервальной группировкой:

$$S = \{I_n: I_n \in \mathbb{N}, n \in \mathbb{N}, 0 < n < K\} \quad (2)$$

где I_n – количество зарегистрированных фотонов в n -ом канале, K – количество интервалов в спектре (Рисунок 1).

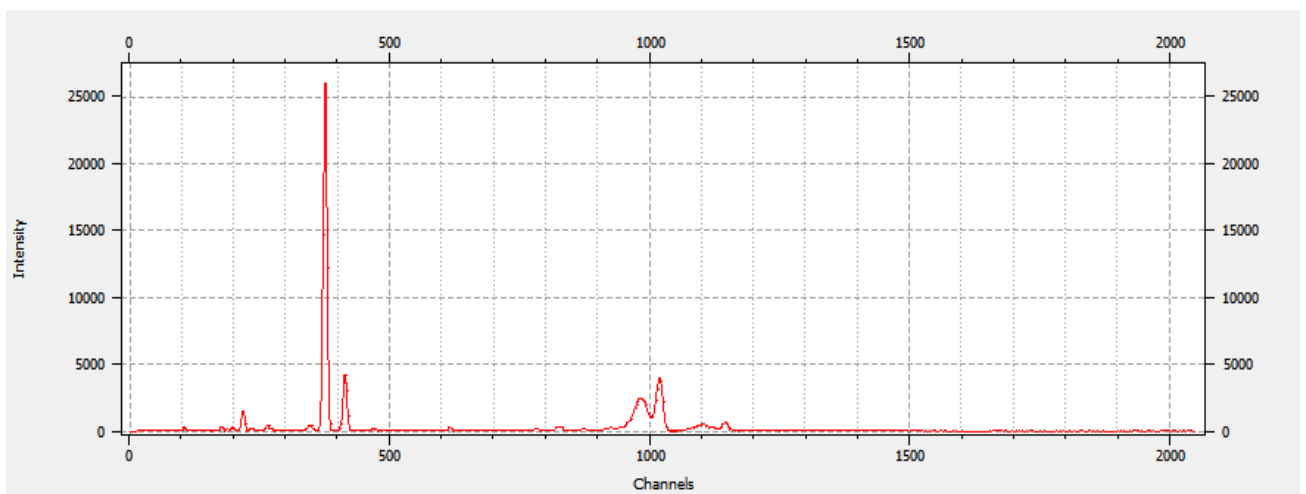


Рисунок 1 - Представление спектра в виде гистограммы

Концентрация химического элемента является отображением G интервального вариационного ряда S на множество положительных вещественных чисел:

$$G: S \rightarrow \mathbb{R}^+. \quad (3)$$

Отображение G – в спектральном анализе называется *градуировкой*.

Существует несколько методов расчета концентраций химических элементов. Одним из самых часто используемых, входящий в состав программного обеспечения почти всех современных спектрометров, является метод корректировки по интенсивности [1, 2].

Установлено, что концентрация химического элемента в легкой, нейтральной матрице, то есть такой, что химические элементы, составляющие эту матрицу никак не влияют на интенсивность рентгеновской флуоресценции определяемого химического элемента, прямо пропорциональна интенсивностям в пиках линий ХРИ этого элемента:

$$C_A = \frac{n_A}{\eta_A}, \quad (4)$$

где C_A – концентрация определяемого химического элемента, n_A – интенсивность в пике линии ХРИ определяемого химического элемента, η_A – удельная скорость счета количества импульсов в этой линии при изменении концентрации определяемого элемента на 1%. Величина η_A рассчитывается через удельные интенсивности в пике линии ХРИ, полученные от образцов с известными концентрациями определяемого химического элемента.

В реальных условиях в подавляющем большинстве случаев на скорость счета η_A оказывают влияние другие химические элементы, входящие в химический состав изучаемого материала. Тяжелые химические элементы, испуская рентгеновские флуоресцентные фотоны высоких энергий, служат дополнительным источником возбуждения для более легких элементов. Легкие элементы являются причиной рассеяния фотонов. В таких случаях рассчитанная

концентрация C_A содержит значительные отклонения от реальных концентраций и, следовательно точность измерения будет неудовлетворительной (Рисунок 2).

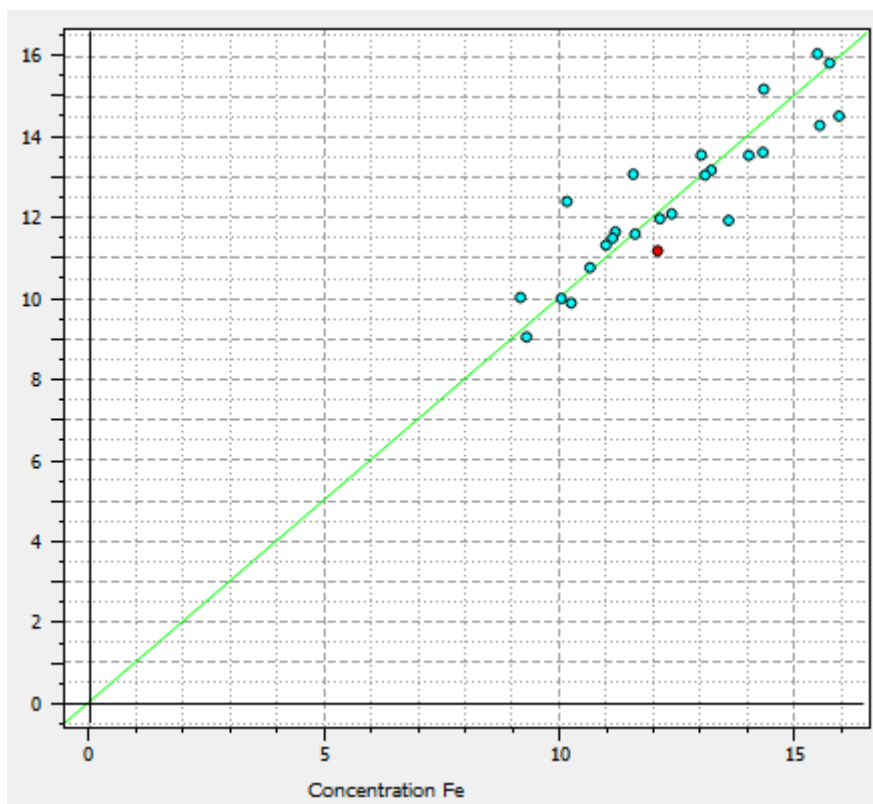


Рисунок 2 - Рассчитанная концентрация Fe без учета влияющих химических элементов

Кроме того, взаимное влияние элементов друг на друга сложно и транзитивно. Как показано на Рисунке 3, это приводит к нелинейной зависимости между интенсивностями в пиках линий ХРИ и концентрацией химического элемента.

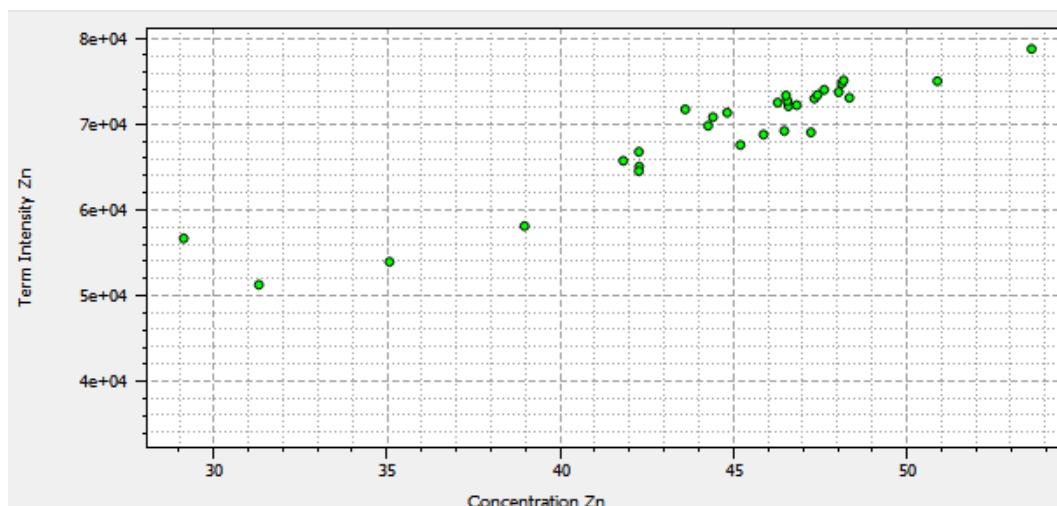


Рисунок 3 - Нелинейная зависимость между интенсивностью в линии ХРИ цинка и его концентрацией в исследуемом материале

Для коррекции влияния других химических элементов в уравнение (4) вводятся дополнительные корректирующие переменные. Ввиду сложного химического состава реальных образцов и сложного характера взаимного влияния одних химических элементов на другие, уравнение концентрации приобретает вид уравнения с многими независимыми переменными, выражающими интенсивности во влияющих пиках, то есть концентрация определяемого химического элемента есть функция от интенсивностей в m пиках энергетических линий:

$$C_A = f(n_1, n_2, \dots, n_m). \quad (5)$$

В большинстве случаев используется уравнение концентрации в виде полиномиальной регрессии:

$$C_A = a_{A0} + a_{A1}n_A + a_{A2}n_A^2 + \sum_{i=1}^m a_{i1}n_i + \sum_{i=1}^m a_{i2}n_i^2 + \sum_{i=1, j=1, j \neq i}^m a_{i3}n_in_j, \quad (6)$$

где C_A – концентрация определяемого химического элемента, n_A - интенсивность в пике линии ХРИ определяемого химического элемента, n_i, n_j - интенсивность в пике линии ХРИ влияющего химического элемента, a_i – коэффициенты, отражающие степень влияния интенсивности в пике соответствующей линии

ХРИ. Учет корректирующих интенсивностей значительно увеличивает точность расчета концентрации (Рисунок 4).

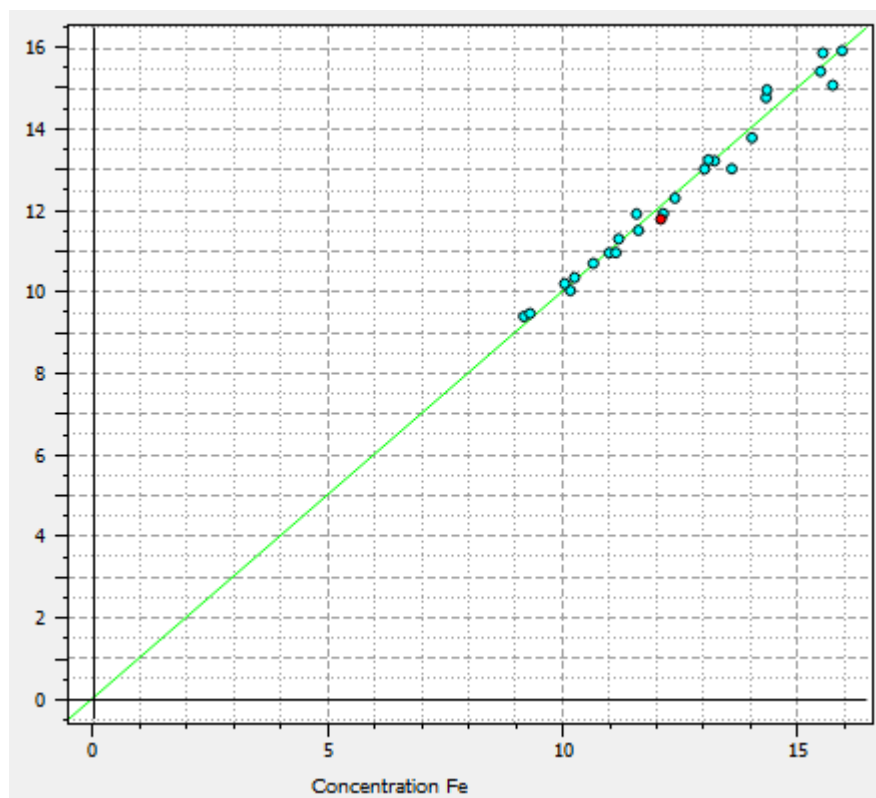


Рисунок 4 - Рассчитанная концентрация Fe с учетом влияющих химических элементов

Независимыми переменными регрессионной модели являются значения интенсивностей характеристического излучения в линиях элемента, концентрацию которого требуется определить, а также значения интенсивностей характеристического излучения в линиях влияющих элементов.

Для создания градуировки производятся следующие операции:

- выделяется базовый пик на спектрах – наиболее выраженный пик линии характеристического излучения определяемого элемента,
- по линиям спектра определяется наличие влияющих химических элементов в исследуемом материале и принимается решение о включении пиков влияющих элементов в градуировку,

- определяется необходимость и вид нормирования значений интенсивностей во включенных пиках,
- определяется необходимость создания искусственных предикторов в регрессионном уравнении, таких как произведение или сумма значений интенсивностей в двух пиках,
- производится расчет регрессионных коэффициентов, осуществляется проверка качества градуировки на градуировочном и проверочном наборах спектров.

Правильный выбор этих параметров критически влияет на качество градуировки и, как следствие, на точность результатов анализа. Процесс создания градуировки является итеративным, так как взаимное влияние элементов друг на друга сложно и транзитивно.

На каждой итерации создается градуировка с различным набором предикторов. Далее из множества созданных градуировок выбирается градуировка с максимальными показателями качества – коэффициента детерминации R^2 и среднеквадратического отклонения σ .

Создание качественной градуировки является наиболее сложной операцией и является одной из проблем, связанных с использованием спектрального анализа.

Для решения этих проблем предлагается для создания градуировок применить алгоритмы глубокого обучения нейронных сетей.

2.2 Теоретическая основа нейронной сети

2.2.1 Структура нейронной сети

Нейронные сети имеют ряд отличительных свойств по сравнению с другими методами, такие как устойчивость к шумам во входных данных, способность к обобщению, возможность аппроксимировать функциональную зависимость любого порядка нелинейности и сложности.

Нейронная сеть представляет собой структуру состоящей из элементарных элементов – искусственных нейронов. Структура искусственного нейрона представлена на Рисунок 5.

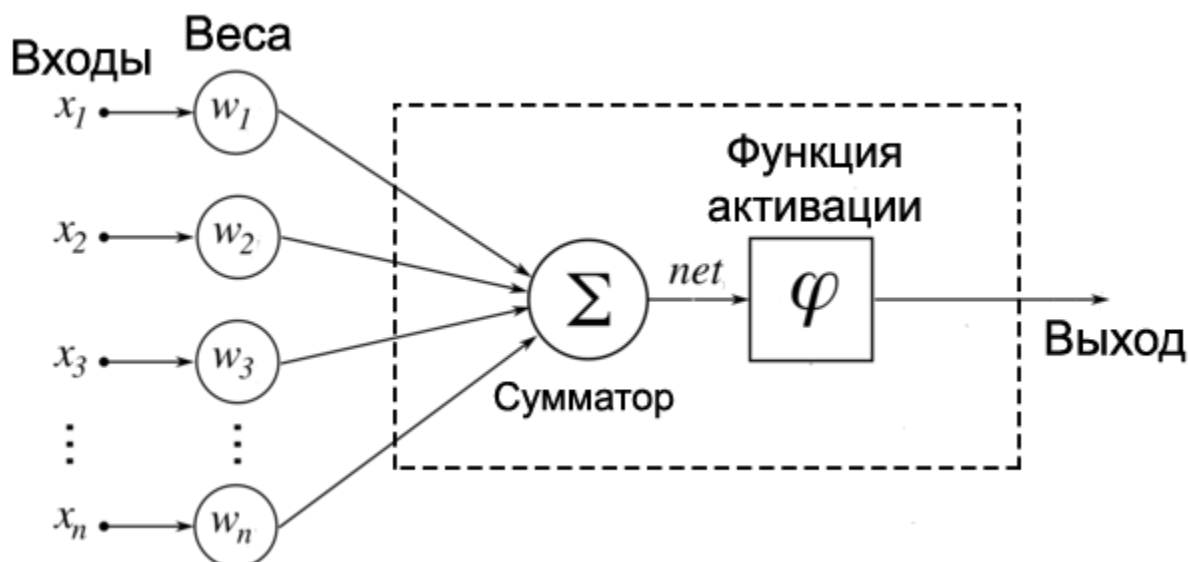


Рисунок 5 - Структура искусственного нейрона

На вход нейрона подается n-мерный вектор некоторых значений X . Каждое значение x_i умножается на весовой коэффициент w_i и поступает на сумматор. Далее сумма взвешенных элементов вектора X поступает на вход функции активации φ .

Таким образом, искусственный нейрон реализует функцию:

$$y = \varphi(u), u = \sum_{i=1}^n x_i w_i, \quad (7)$$

где φ – функция активации, u – взвешенная сумма, как сумма n произведений входных сигналов на соответствующие веса.

При линейной функции активации функция (7) эквивалентна функции линейной регрессии. Для реализации многих практических задач этого недостаточно, так как зависимости имеют, как правило нелинейный характер.

По теореме Колмогорова (1957 г.) «Любая непрерывная функция от n переменных может быть представлена в виде

$$F(x_1, x_2, \dots, x_n) = \sum_{j=1}^{2n+1} g_j \left(\sum_{i=1}^n \varphi_{ij}(x_i) \right), \quad (8)$$

где g_j и φ_{ij} непрерывные функции, причем φ_{ij} не зависят от функции F [23].

По теореме (8) последовательным соединением достаточного количества нейронов в сеть, то есть суперпозицией реализуемых ими функций, можно реализовать сколь угодно сложную зависимость с любой точностью.

Обычно для решения регрессионных задач используется полносвязные нейронные сети с одним или большим количеством скрытых слоёв (Рисунок 6).

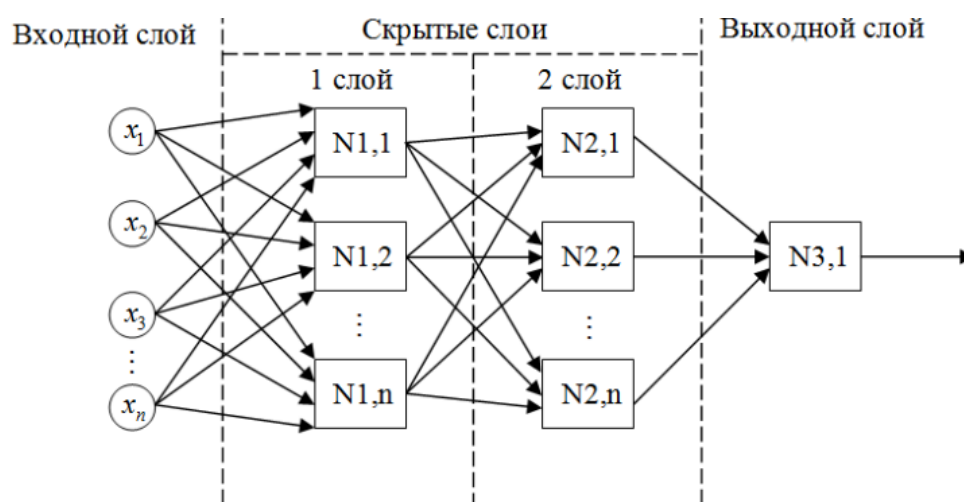


Рисунок 6 - Структурная схема многослойной полносвязной нейронной сети

На вход нейронной сети подается вектор признаков X , описывающий объект, являющиеся предикторами для регрессионной модели. Выходной слой нейронной сети - один нейрон, на выходе которого значение регрессионной функции y , реализованной нейронной сетью. Реализуемая нейронной сетью функция y определяется функциями активации нейронов $\varphi_i(u)$ и весовыми коэффициентами w_{ji} .

Функция активации нейронов выбирается на этапе проектирования нейронной сети в зависимости от задачи. Для регрессионных задач функция активации, как правило, линейная:

$$f(x) = x, \quad (9)$$

либо ReLu:

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \quad (10)$$

Значения весовых коэффициентов настраиваются в процессе обучения нейронной сети.

2.2.2 Обучение нейронной сети

Для обучения нейронной сети необходимы множество векторов признаков объектов

$$X = \{x_1, x_2, \dots, x_l\} \quad (11)$$

и множество соответствующих каждому объекту векторов желаемых откликов нейронной сети

$$Y = \{y_1, y_2, \dots, y_l\} \quad (12)$$

где l – количество объектов выборке.

Совокупность пар

$$X^l = (x_i, y_i)^l \quad (13)$$

называется *обучающая выборка*.

Алгоритм обучения состоит в последовательной подаче на вход нейронной сети вектора признаков объекта x_i и подстройкой весовых коэффициентов таким образом, чтобы вектор значений на выходе нейронной сети $a(x_i)$ был как можно ближе вектору желаемых откликов y_i .

Для определения близости векторов вводится функция потерь $L(x_i, w)$, где w – вектор весовых коэффициентов нейронной сети. Для задач регрессии в качестве функции потерь чаще всего используют абсолютное значение ошибки

$$L(x_i, w) = |a(x_i, w) - y_i| \quad (14)$$

либо квадратичную ошибку

$$L(x_i, w) = (a(x_i, w) - y_i)^2 \quad (15)$$

Для оценки качества нейронной сети вводится функционал качества – эмпирический риск:

$$Q(X^l, w) = \frac{1}{l} \sum_{i=1}^l L(x_i, w) \quad (16)$$

Задача обучения сводится к задаче оптимизации эмпирического риска:

$$\mu(X^l) = \underset{w}{\operatorname{argmin}} Q(X^l, w). \quad (17)$$

Оптимизация производится методом обратного распространения ошибки:

1. веса w иницируются случайными небольшими значениями;
2. из обучающей выборки случайным образом выбирается пара (x_i, y_i) ;
3. на вход нейронной сети подается вектор признаков x_i ;
4. вычисляется локальный градиент $\delta_j = L(x_i, w) \cdot \frac{df_j(e)}{de} \cdot w_{ij}$;
5. производится коррекция весов нейронной сети по направлению от выхода ко входу по формуле

$$\Delta w_{ij} = -\eta \delta_j o_i, \quad (18)$$

где o_i – выход i -го нейрона, $f_j(e)$ – функция активации j -го нейрона.

Шаги 2-5 повторяются до тех пор пока веса w нейронной сети не стабилизируются.

Таким образом, нейронная сеть с настроенными весовыми коэффициентами и правильно подобранной функцией активации реализует функционал любого вида отображающего входной вектор признаков в выходное – целевое значение.

2.2.3 Регуляризация параметров нейронной сети

Одна из проблем нейронных сетей заключается в их склонности к переобучению. Переобучение – это такой эффект, когда нейронная сеть очень точно аппроксимирует исходную функцию в точках пространства признаков, соответствующим объектам обучающей выборки. В этом случае нейронная сеть

учитывает в своей модели даже шумовую дисперсию признаков из-за чего модель приобретает высокий порядок нелинейности. Но в интервалах между этими точками аппроксимирующая нейронной сетью функция может проходить непредсказуемым образом. Это приводит к тому, что на обучающих данных ошибка нейронной сети очень мала, в то время как на данных, не вошедших в обучающую выборку, ошибка нейронной сети может быть неприемлемой. На Рисунке 7 представлен эффект переобучения нейронной сети.

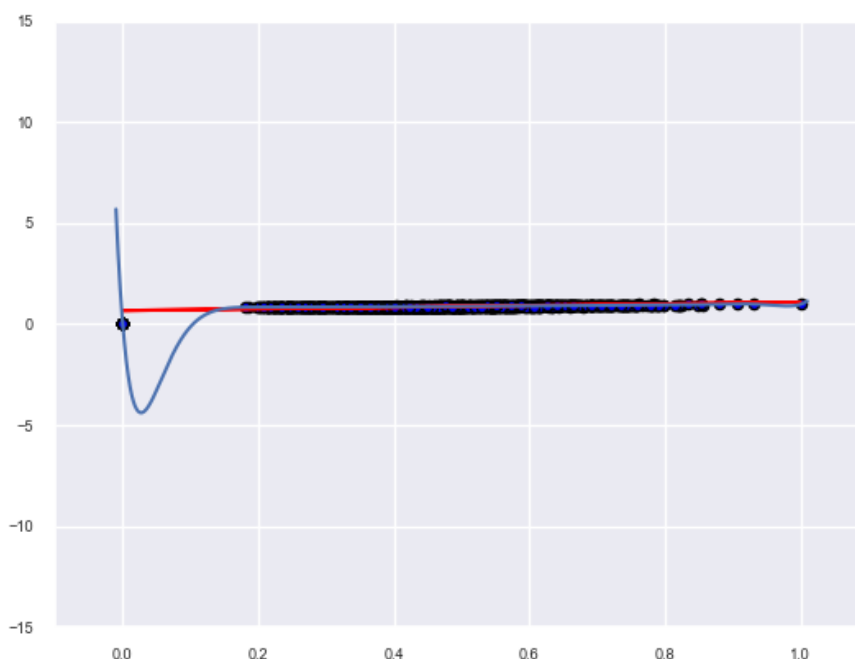


Рисунок 7 – Эффект переобучения нейронной сети. Синяя линия – аппроксимирующая функция нейронной сети

Данные обучающей выборки на Рисунке 7 распределены неравномерно. Поэтому в интервале $[0, 0,2]$ между двумя группами данных аппроксимирующая функция даёт значительную ошибку.

Для минимизации эффекта переобучения используются методы регуляризации параметров. Эти методы заключаются в таком изменении функционала качества $Q(X^l, w)$, чтобы он учитывал усложнение модели как фактор, ухудшающий её качество. Для нейронных сетей используется L_1 – регуляризация:

$$Q_{L_1} = Q(X^l, w) + \lambda_1 \sum_{i=1, j=1}^{N, K} |w_{ij}| \quad (19)$$

и L_2 – регуляризация:

$$Q_{L_2} = Q(X^l, w) + \lambda_2 \sum_{i=1, j=1}^{N, K} (w_{ij})^2, \quad (20)$$

где λ_1 и λ_2 – регуляризационные гиперпараметры.

Также может использоваться комбинация L_1 и L_2 регуляризаций.

Регуляризация предполагает, что все входные признаки имеют одинаковое влияние на результирующий отклик модели. На практике различные признаки могут сильно отличаться по значениям их дисперсий и по средним значениям обучающей выборки. В таких случаях входные данные должны быть стандартизованы или нормированы.

Значения гиперпараметров λ_1 и λ_2 подбираются экспериментально на этапе проектирования нейронной сети.

2. Разработка программного продукта

2.1 Решение проблемы выбора признаков

Использование полносвязных нейронных сетей для решения задач регрессии является широко распространенным. Но для создания градуировки для спектрометра без предварительной обработки спектров применение нейронной сети имеет проблемы. Одна из проблем – проблема выбора признаков.

На Рисунке 8 представлены два пика линий ХРИ рентгенофлуоресцентного спектра.

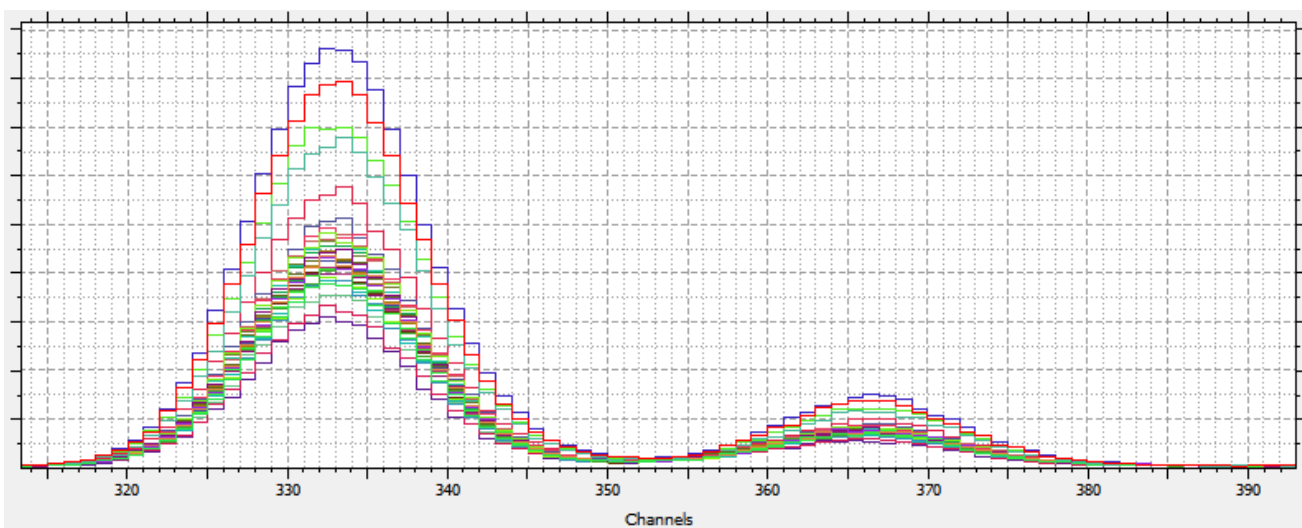


Рисунок 8 - Мультиколлинеарность значений интенсивностей в каналах, представляющих один пик линии ХРИ

Каждый пик представлен несколькими десятками каналов: первый пик – каналы 318-350, второй – каналы 354-380. Очевидно, что значения интенсивностей в каналах каждого пика имеют сильную мультиколлинеарность. Поэтому регрессионная модель, предикторами которой будут являться значения интенсивностей в каналах спектра будет обладать высокой статистической неопределенностью. Оценки такой модели будут обладать неприемлемым значением дисперсии.

В результате исследовательских работ для решения этой проблемы предложен и реализован метод создания градуировок с использованием сверточной нейронной сети.

Сверточные нейронные сети имеют широкое распространение в алгоритмах распознавания образов на изображениях.

Свертка представляет собой функцию взаимной корреляции двух функций. Одна из функций – исходный сигнал $X = \{x_1, x_2, \dots, x_l\}$, другая функция w_a , где $a = -A, \dots, +A$, – фильтрующая. Во время операции свертки фильтрующая функция w «пробегает» по всему входному сигналу и на каждой позиции вычисляется скалярное произведение

$$(x * w)[i] = \sum_{a=-A}^{+A} w_a x[i + a]. \quad (21)$$

В результате позиции в функции свёртки, имеющие максимальные значения, указывают на позиции в исходном сигнале, наиболее схожие с фильтрующей функцией (Рисунок 9).

X	0	1	1	1	0	0	1	0	1	0
w							1	0	1	
Y	1	2	1	1	1	0	2	0		

Рисунок 9 – Схема операции свертки.

В свёрточных нейронных сетях операция свёртки используется для нахождения паттернов, распределенных в векторе признаков входного сигнала. Сигнал во входном слое сканируется несколькими свёрточными фильтрами. Сигналы свертки подаются на нейроны следующего слоя. Каждый фильтр образует свою отдельную группу нейронов в следующем слое, называемую картой признаков. Каждая карта признаков отображает наличие в исходном сигнале определенного признака. Значимые величины в карте признаков показывают наличие этого признака в определенной позиции входного сигнала. Далее, каждая карта признаков сканируется следующей группой фильтров, выделяющих в них абстракции и обобщения ещё более высоких уровней.

Фильтры представляют собой наборы весовых коэффициентов, настраиваемые во время обучения сверточной нейронной сети.

Кроме непосредственных сверток в свёрточной нейронной сети производится операция пуллинга или изменения масштаба. Как и при операции свёртки, карта признаков сканируется окном некоторого размера с таким шагом, чтобы каждое новое сканирование не пересекалось в предыдущим. Из карт признаков выбирается одно значение, как правило, максимальное. Таким образом, окно размером в два значения уменьшает масштаб в два раза. Это позволяет анализировать изображение на крупных масштабах. Кроме того

уменьшается размерность нейронной сети, что позволяет уменьшить количество настраиваемых параметров, что ускоряет обучение. Схема операции пуллинга представлена на Рисунке 10.

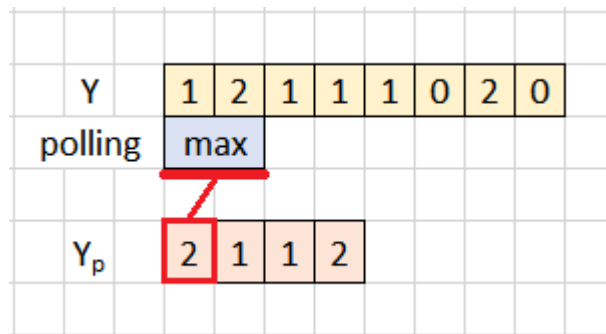


Рисунок 10 – Схема операции пуллинга

В свёрточной нейронной сети, в конце последовательности свёрточных и пуллинговых слоёв находятся полносвязные нейронные слои. Признаки, выделенные из входного сигнала свёрточными слоями, выравниваются в специальном слое и подаются в полносвязные нейронные слои, в которых производится конечная операция, как правило, классификация объектов, присутствующих на изображении. Схема свёрточной нейронной сети для классификации изображений представлена на Рисунке 11.

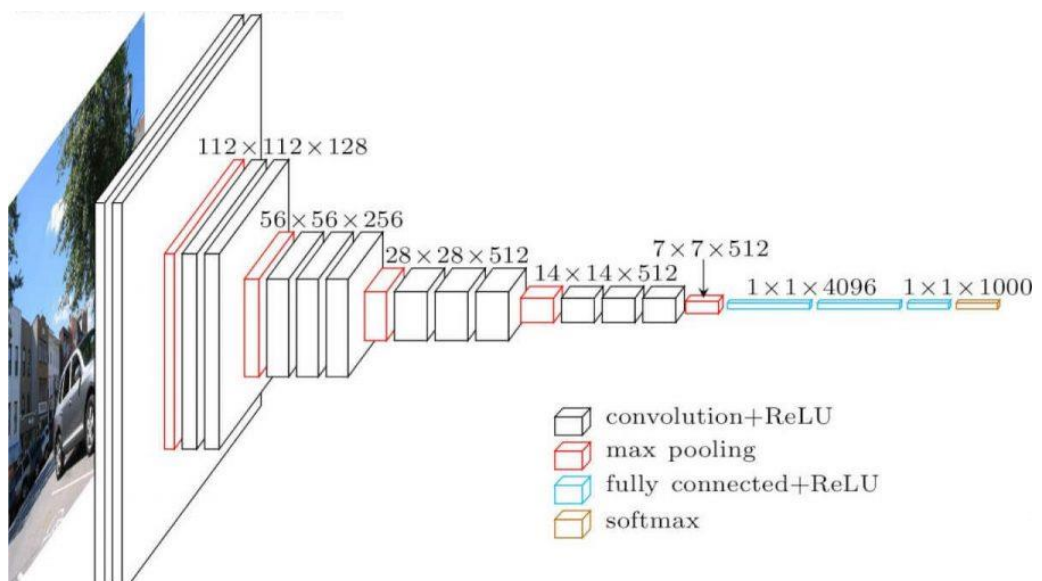


Рисунок 11 - Схема свёрточной нейронной сети VGG16

Рентгеновский спектр можно представить в виде одномерного изображения. Отдельные пики линий ХРИ являются крупномасштабными элементами этого изображения. При помощи сверточных слоев эти элементы извлекаются из спектра как признаки, более высокого уровня абстракции чем отдельные каналы, и затем подаются на вход полностью связанной – регрессионной части нейронной сети. Регрессионная часть имеет один выходной нейрон с функцией активации ReLu, подходящей для задачи регрессии. Таким образом решается проблема мультиколлинеарности входных признаков.

2.2 Решение проблемы малой выборки

Вторая проблема применения нейронной сети для задач градуировки спектрометра связана с малым размером обучающей выборки.

Для качественного обучения нейронных сетей требуется сотни и тысячи объектов в обучающей выборке. Как правило, количество спектров, доступных для создания градуировки не превышает нескольких десятков. Так как спектры одного и того же материала отснятые на разных спектрометрах значительно отличаются друг от друга в силу индивидуальных различий в параметрах компонентов аппаратуры спектрометра, таких как рентгеновская трубка, полупроводниковый детектор рентгеновского излучения, измерительная камера и др., для градуировки допускается использование только собственных для каждого спектрометра спектров. Использование же столь малого размера обучающей выборки приводит к неудовлетворительному качеству градуировочной модели.

В ходе исследования для решения этой проблемы предложен двухэтапный метод обучения нейронной сети.

На первом этапе производится предварительное обучение нейронной сети на наборе спектров собранном из спектров измеренных для определения всех возможных химических элементов и соединений для всех возможных диапазонов концентраций. На этом этапе нейронная сеть обучается

конструировать и выбирать важные признаки из сырого спектра. Интенсивности в каналах спектра объединяются в абстракции более высоких порядков, соответствующих значениям интенсивностей в пиках линий ХРИ, интенсивностей пиков рассеяния и других присутствующих в спектре артефактов.

На втором этапе производится дообучение нейронной сети на градуировочном наборе для целевого химического элемента и диапазона концентраций.

2.3 Выбор необходимых инструментальных средств

Для проведения исследования на спектрометре СРВ-1В производства ТОО «ТехноАналит» были измерены градуировочные спектры различных материалов с широким диапазоном концентраций химических элементов. Некоторые наборы градуировочных спектров представлены на Рисунке 12.



Рисунок 12 - Наборы градуировочных спектров

Для реализации тестовой версии программного продукта и вычислительных экспериментов был выбран язык программирования Python 3.6 и среда программирования Jupyter Lab. Нейронная сеть реализована с помощью

библиотеки Keras. Обучение нейронной сети проводилось с помощью графической карты RTX 2070 и технологии CUDA, позволяющей использовать для обучения нейронных сетей ядра графического процессора.

Для интеграции программного продукта в существующее программное обеспечение ТОО «ТехноАналит» были использованы следующие инструментальные средства:

- Язык программирования C++,
- Кроссплатформенный фреймворк Qt 5.15.0,
- IDE Qt Creator,
- Интегрированный в Qt Creator редактор UML моделей,
- Doxygen – инструментальное средство для автоматической генерации документации проекта,
- Система контроля версий Git.

Данный инструментальный набор является внутренним стандартом «ТОО ТехноАналит» для разработки программного обеспечения. Программные продукты, разработанные с использованием этого набора, удовлетворяют требованиям кроссплатформенности и производительности.

3. Реализация программного продукта

3.1 Тестовая версия программного продукта

Реализация тестовой версии программного продукта выполнена на языке Python 3.6 в среде Jupyter Lab. Для реализации нейронной сети использовалась библиотека Keras. Keras является прикладным интерфейсом приложения для библиотеки TensorFlow, поддерживающей обучение нейронных сетей с использованием ядер графического процессора видеокарты компьютера, что позволяет значительно сократить время обучения нейронной сети.

Для контроля процесса обучения выводились графики, отображающие значения функции потерь для обучающей и проверочной выборок для каждой эпохи обучения (Рисунок 13).

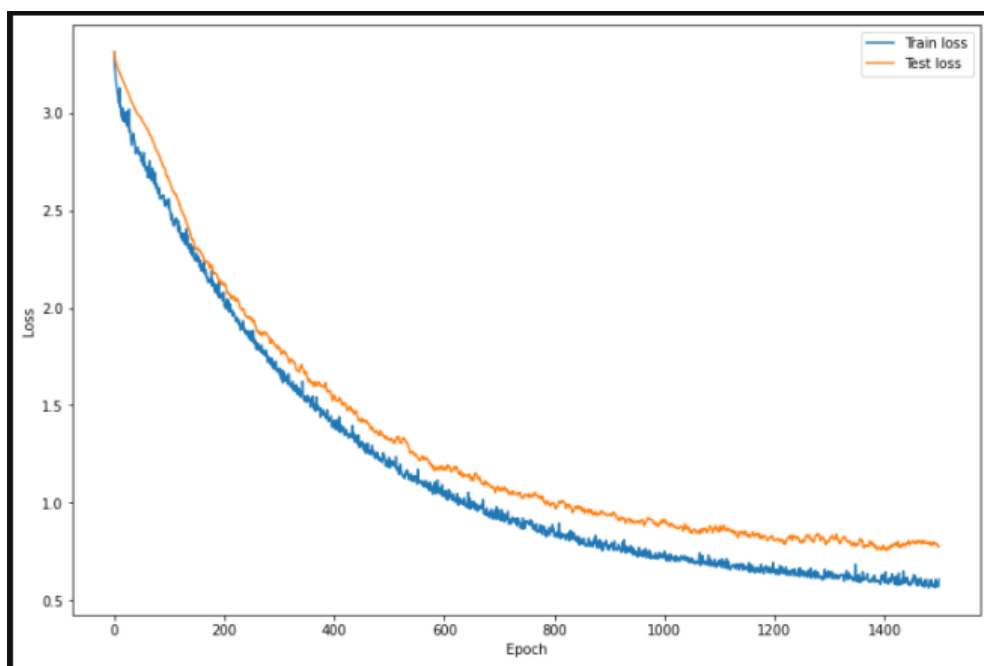


Рисунок 13 – Отображение функции потерь для обучающей и проверочной выборок в процессе обучения нейронной сети

При разработке программного продукта использовалась система контроля версий git. Тестовая версия программного продукта преследует цели:

- оптимизации архитектуры нейронной сети;
- оптимизации значений её гиперпараметров;
- проверки методики обучения;
- проверки качества создаваемых градуировок.

Для проведения вычислительных экспериментов градуировочные спектры были распределены по папкам и дополнены файлами с химическими концентрациями элементов в формате *.csv (Рисунок 14).

Pb	Zn	Cu	Fe
1.93	32.69	4.56	19.2
1.3	31.3	5.74	20.06
1.92	35.07	6.8	16.87
4.32	42.28	3.43	11.83
3.58	45.2	3.74	10.21
3.52	47.24	3.51	9.02
2.67	45.87	3.72	10.54
0.8	53.61	2.06	6.38
3.51	38.97	6.28	12.53
4.35	42.27	3.16	11.7
2.16	42.27	3.83	12.2
1.03	50.89	3.14	7.87
2.87	44.41	1.82	11.6
2.46	46.28	2.07	10.74
2.96	46.59	2.1	10.45
2.34	48.04	2.06	9.61
2.05	46.56	2.08	10.91
1.63	44.82	2.4	12.07
1.26	44.27	2.76	12.05

Рисунок 14 – Фрагмент файла .csv с химическими концентрациями элементов

3.2 Архитектура нейронной сети

В качестве отправной точки для нахождения оптимальной архитектуры нейронной сети была выбрана архитектура VGG16. Далее, в архитектуре производились небольшие изменения и затем проверка параметров качества нейронной сети. В случае ухудшения, изменения отменялись и нейронная сеть возвращалась к исходной архитектуре. В случае улучшения – производились дальнейшие изменения в этом же направлении до стабилизации параметров качества нейронной сети.

Конфигурация разработанной нейронной сети представлена на Рисунке 15. Гиперпараметры разработанной нейронной сети представлены на Рисунке 16.

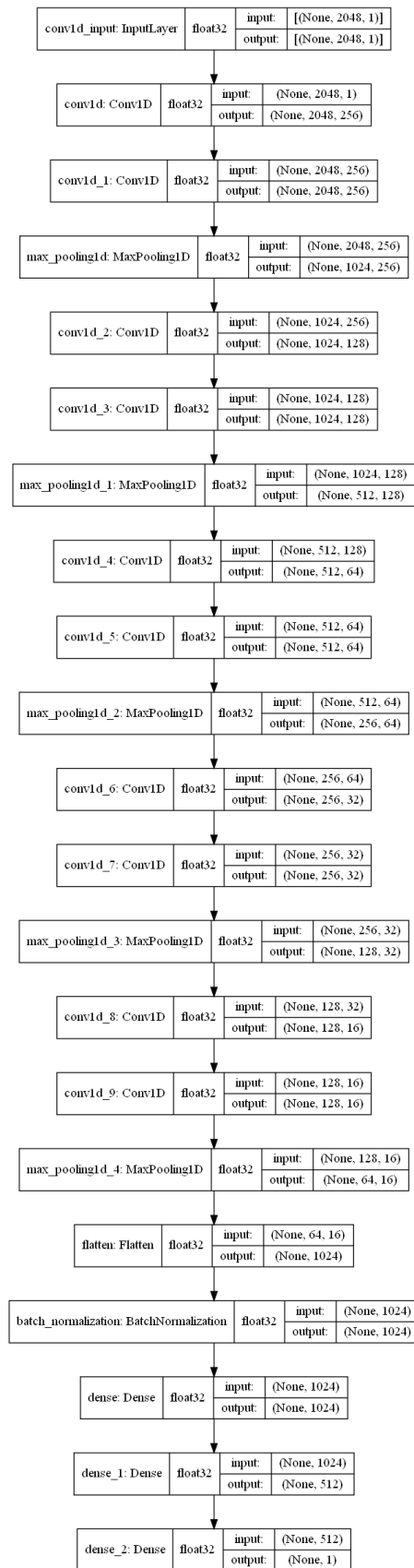


Рисунок 15 – Конфигурация разработанной нейронной сети

Model: "sequential"

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 2048, 256)	1024
conv1d_1 (Conv1D)	(None, 2048, 256)	196864
max_pooling1d (MaxPooling1D)	(None, 1024, 256)	0
conv1d_2 (Conv1D)	(None, 1024, 128)	98432
conv1d_3 (Conv1D)	(None, 1024, 128)	49280
max_pooling1d_1 (MaxPooling1D)	(None, 512, 128)	0
conv1d_4 (Conv1D)	(None, 512, 64)	24640
conv1d_5 (Conv1D)	(None, 512, 64)	12352
max_pooling1d_2 (MaxPooling1D)	(None, 256, 64)	0
conv1d_6 (Conv1D)	(None, 256, 32)	6176
conv1d_7 (Conv1D)	(None, 256, 32)	3104
max_pooling1d_3 (MaxPooling1D)	(None, 128, 32)	0
conv1d_8 (Conv1D)	(None, 128, 16)	1552
conv1d_9 (Conv1D)	(None, 128, 16)	784
max_pooling1d_4 (MaxPooling1D)	(None, 64, 16)	0
flatten (Flatten)	(None, 1024)	0
batch_normalization_1 (Batch Normalization)	(None, 1024)	4096
dense_3 (Dense)	(None, 1024)	1049600
dense_4 (Dense)	(None, 512)	524800
dense_5 (Dense)	(None, 1)	513

Total params: 1,973,217

Trainable params: 1,971,169

Non-trainable params: 2,048

Рисунок 16 – Гиперпараметры разработанной нейронной сети

В свёрточной части нейронной сети использованы одномерные свёрточные слои. В них происходит свертка одномерного изображения спектра с матрицами фильтров. Количество фильтров и, соответственно, карт признаков в слоях уменьшается с 256 до 16. Между слоями происходит операция пуллинга MaxPolling, целью которого является уменьшение размерности. Таким образом

происходит извлечение информации из исходного спектра и обобщение их в картах признаков. Перед полностью связанной частью сети происходит выравнивание карт признаков в слое Flatten, далее в слое BatchNormalization производится пакетная нормализация, то есть распределение значений признаков из выравнивающего слоя Flatten приводятся к стандартному распределению $N(0,1)$. В полностью связанной части нейронной сети используется три слоя с количествами узлов 1024, 512 и 1, соответственно.

При обучении нейронной сети был использован метод кросс-валидации. Градуировочный набор спектров случайным образом разбивался на 5 поднаборов. Далее проводилось 5 итераций обучения, при которых обучающая выборка формировалась из 4 поднаборов, а оставшийся поднабор использовался как тестовый. На каждой итерации наборы менялись.

Для предотвращения эффекта переобучения в первом слое полностью связанной части нейронной сети была использована L1 и L2 регуляризации. Оба алгоритма проявили себя хорошо, обучение значительно стабилизировалось, эффекты переобучения нейронной сети не проявлялись.

Важной частью разработки нейронной сети является подбор её гиперпараметров, таких как функции потерь, скорости обучения, количества нейронов в слоях, количества фильтров в сверточных слоях и т.д. Правильный выбор набора гиперпараметров критически важен для обеспечения сходимости и стабильности модели при обучении, а также её точности. Так как алгоритмов подбора гиперпараметров не существует, подбор осуществляется экспериментально.

Основное внимание было уделено выбору функции потерь. Для задач регрессии в качестве функции потерь используются квадратичная ошибка mse и абсолютная ошибка mae.

В результате вычислительного эксперимента функция потерь – абсолютная ошибка показала несколько лучше, чем среднеквадратическая ошибка (Рисунок 17).

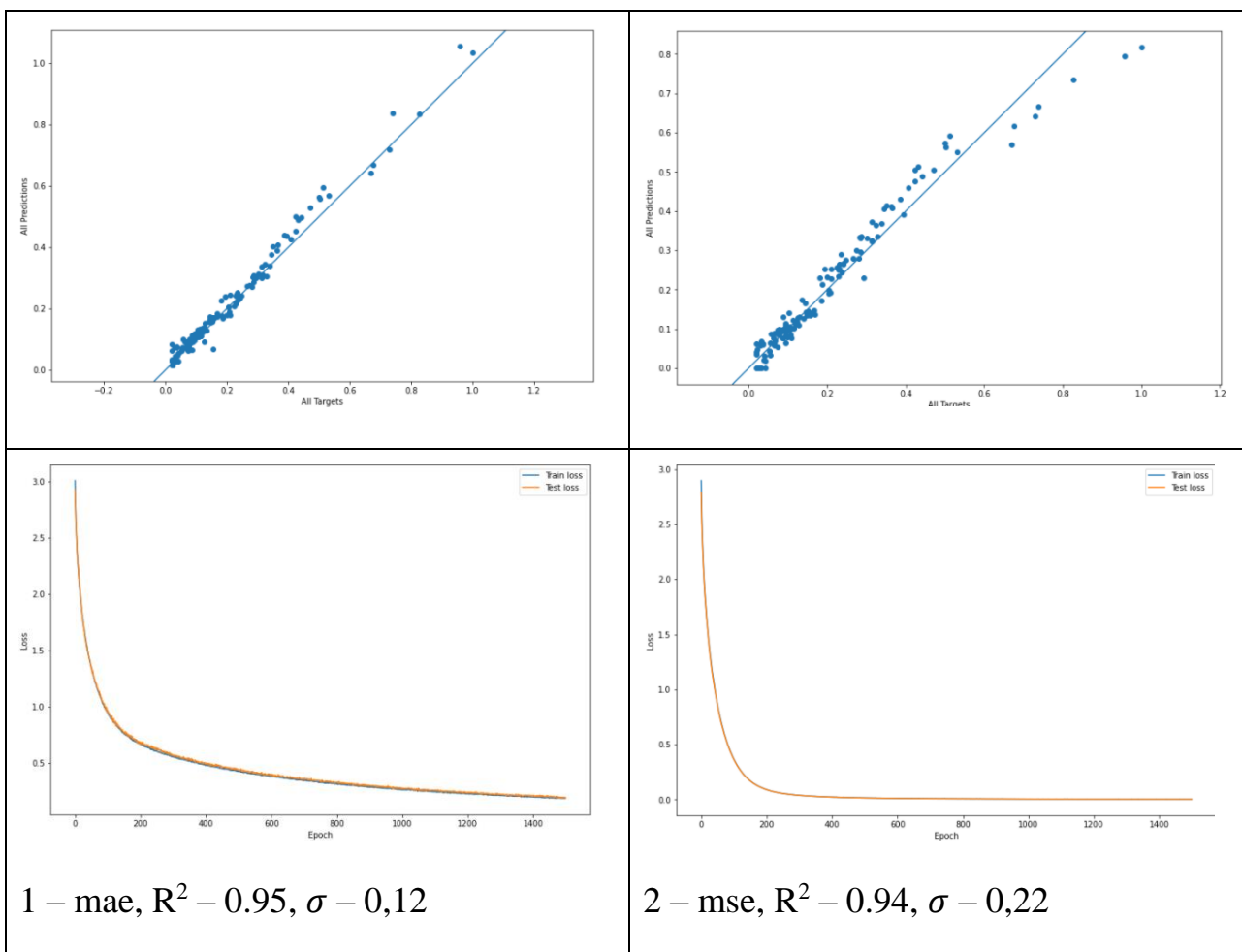


Рисунок 17 – Процесс обучения градуировки для железа Fe. 1 – с функцией потерь – абсолютной ошибкой, 2 – с функцией потерь – среднеквадратической ошибкой

Для ускорения времени обучения во время подбора гиперпараметров были использованы искусственные наборы данных, представляющие единичный пик спектра. В качестве целевого значения использована суммарная интенсивность ХРИ в каналах пика.

Программный код тестовой версии продукта на языке Python представлен в Приложении Б.

3.3. Интеграция программного продукта в программное обеспечение спектрометра СРВ-1В

Спектрометр СРВ – 1В укомплектован программным обеспечением SRVM необходимым для создания градуировок. Программное обеспечение

SRVM создано на языке C++ с использованием библиотеки Qt. На Рисунке 18 представлен интерфейс пользователя программного обеспечения «SRV Methodist».

В программном обеспечении «SRV Methodist» реализован классический градуировочный метод поправок по интенсивностям. После интервью со специалистами ТОО «ТехноАналит» и консультацией с пользователями спектрометра CPB – 1B было решено интегрировать программный продукт в программное обеспечение «SRV Methodist» в виде программного модуля. Такое решение позволяет в одном приложении создавать градуировки различными методами для сравнения их качества.



Рисунок 18 – Программное обеспечение SRV Methodist в действии

В ходе работ по сбору и анализу требований к программному модулю должны быть определены и формально описаны действующие лица, их цели (Таблица 1), функциональные (Таблица 2) и нефункциональные требования

(Таблица 3), диаграмма вариантов использования (Рисунок 19), сценарии использования (Таблицы 4-5).

Таблица 1 - Действующие лица и их цели

Действующее лицо	Цель
Оператор спектрометра	Получение файла градуировки в формате *.clbx, совместимого с программным обеспечением ТОО «ТехноАналит».

Таблица 2 - Функциональные требования

1	Загрузка спектральных данных из файлов спектров *.spe
2	Расчет градуировки по загруженным данным спектров
3	Сохранение рассчитанной градуировки в файл *.clbx
4	Возможность перезаписи градуировки в файле *.clbx

Таблица 3 - Нефункциональные требования

1	Загрузка предварительно обученной свёрточной части нейронной сети из файла.
3	Возможность использования на ОС Windows и Linux.

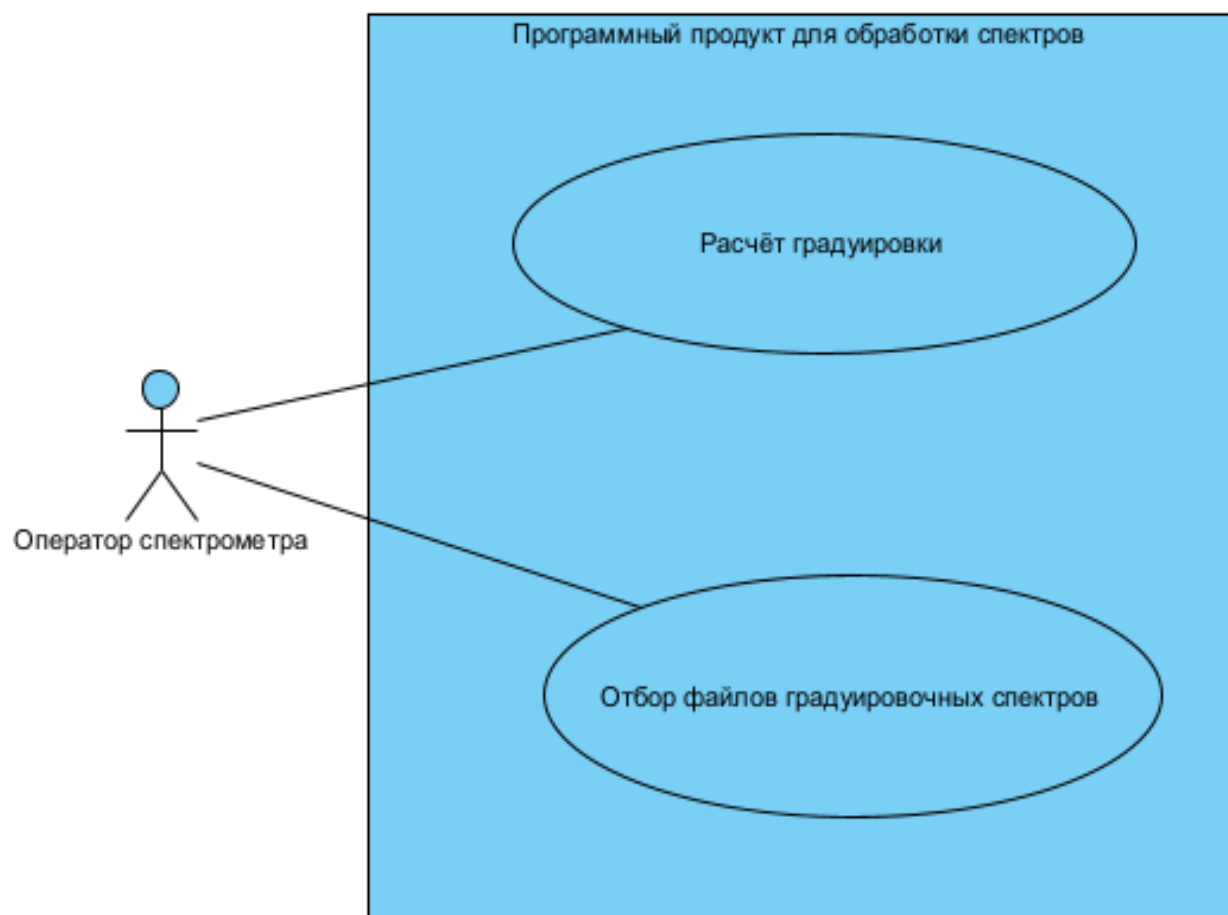


Рисунок 19 - Диаграмма вариантов использования программного модуля

Таблица 4 – Сценарий использования «Расчет градуировки»

Сценарий использования UC1	Расчет градуировки
Действующие лица	Оператор спектрометра, программный модуль
Цель	Получить градуировку рассчитанную разработанным программным модулем
Предусловие	В программу «SRV Methodist» загружен набор градуировочных спектров
<p>Успешный сценарий:</p> <ol style="list-style-type: none"> 1. Оператор спектрометра активирует расчет градуировки. 2. Программный продукт загружает из файла свёрточную часть нейронной сети, загружает данные спектров из приложения «SRV Methodist», запускает процесс обучения нейросети. 3. Программный модуль сохраняет рассчитанную градуировку в файл *.clbx. 	
<p>Расширения:</p> <ol style="list-style-type: none"> 2а. В стандартном месте хранения файл свёрточной нейронной сети не обнаружен. 2а1. Происходит обучение всей нейронной сети. 	
Результат	Оператор спектрометра получил градуировку в виде файла *.clbx.

Основной функционал программного модуля инкапсулирован в класс `ZCnnCalibrator`. В классе определена основная функция `zp_calculateCalibration`, которая запускает процедуру создания и обучения нейронной сети. В программном обеспечении «SRV Methodist» градуировка была представлена классом `ZCalibration`. Класс инкапсулирует весь необходимый функционал, необходимый для работы градуировки методом корректировки по интенсивностям. Для того, чтобы программное обеспечение «SRV Methodist» и другие приложения ТОО «ТехноАналит» могли работать с градуировками на основе нейронных сетей, общий необходимый функционал вынесен в новый абстрактный класс `ZAbstractCalibration`. Классы `ZCalibration` и `ZCnnCalibration` наследуют этот класс. Весь код, который ранее использовал класс `ZCalibration` изменен на использование абстрактного класса `ZAbstractCalibration`. Теперь новый класс `ZCnnCalibration` может быть использован в любом месте. Где ранее использовался только класс `ZCalibration`. Соответствующая диаграмма классов приведена на Рисунке 20.

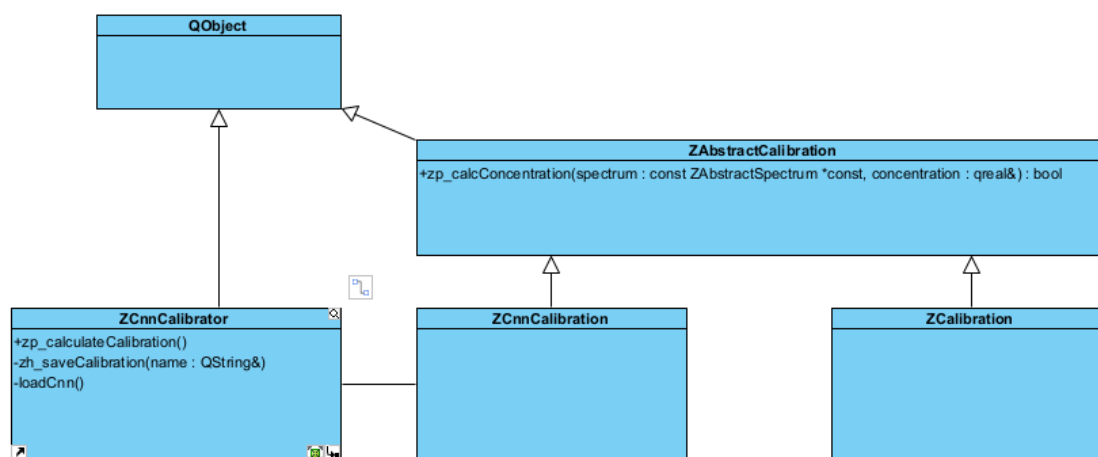


Рисунок 20 – Диаграмма классов интегрируемого программного модуля.

3.4 Реализация интегрального программного модуля.

Реализация программного продукта осуществлялась в IDE Qt Creator (Рисунок 21).

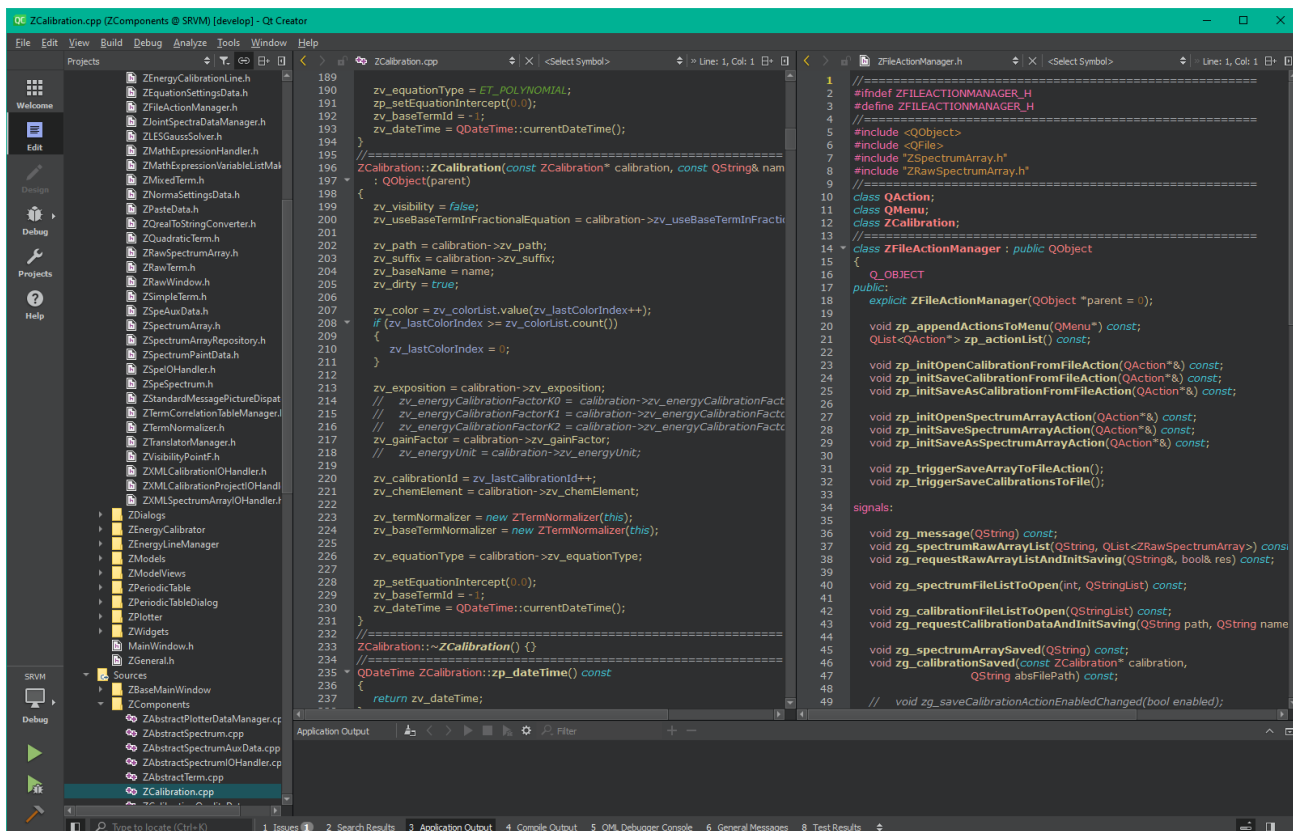


Рисунок 21 - Разработка программного продукта в IDE Qt Creator

При разработке была активно использована система контроля версий git. Для автоматизации операций git использованы скрипты gitFlow.

Интерфейс программного обеспечения для работы с разрабатываемым модулем не менялся, за исключением пункта в меню «Действия – Пересчитать градуировку» (Рисунок 22).

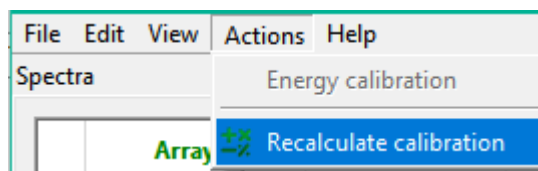


Рисунок 22 – Пункт меню запускающий расчет градуировки при помощи нейронной сети

Набор градуировочных спектров загружается в модуль из текущего массива спектров, открытых в программном обеспечении «SRV Methodist» (Рисунок 23). Спектры со неотмеченным флажком в градуировке не используются.

	File	Spectrum	Pb	Zn
1	<input type="checkbox"/> ZnCt5001.spe		1.93	32.69
2	<input checked="" type="checkbox"/> ZnCt5002.spe		1.3	31.3
3	<input checked="" type="checkbox"/> ZnCt5003.spe		1.92	35.07
4	<input checked="" type="checkbox"/> ZnCt5004.spe		4.32	42.28
5	<input checked="" type="checkbox"/> ZnCt5005.spe		3.58	45.2
6	<input checked="" type="checkbox"/> ZnCt5006.spe		3.52	47.24
7	<input checked="" type="checkbox"/> ZnCt5007.spe		2.67	45.87
8	<input checked="" type="checkbox"/> ZnCt5008.spe		0.8	53.61
9	<input checked="" type="checkbox"/> ZnCt5009.spe		3.51	38.97
10	<input checked="" type="checkbox"/> ZnCt5010.spe		4.35	42.27

Рисунок 23 – Спектры используемые в создании градуировки

После всех необходимых дополнений и изменений в коде программное обеспечение «SRV Methodist» было перекомпилировано.

Для автоматической генерации документации, при выпуске нового релиза программного продукта было использовано инструментальное средство Doxygen. Документация была сгенерирована в формате HTML. Фрагмент документации представлен на Рисунке 24.

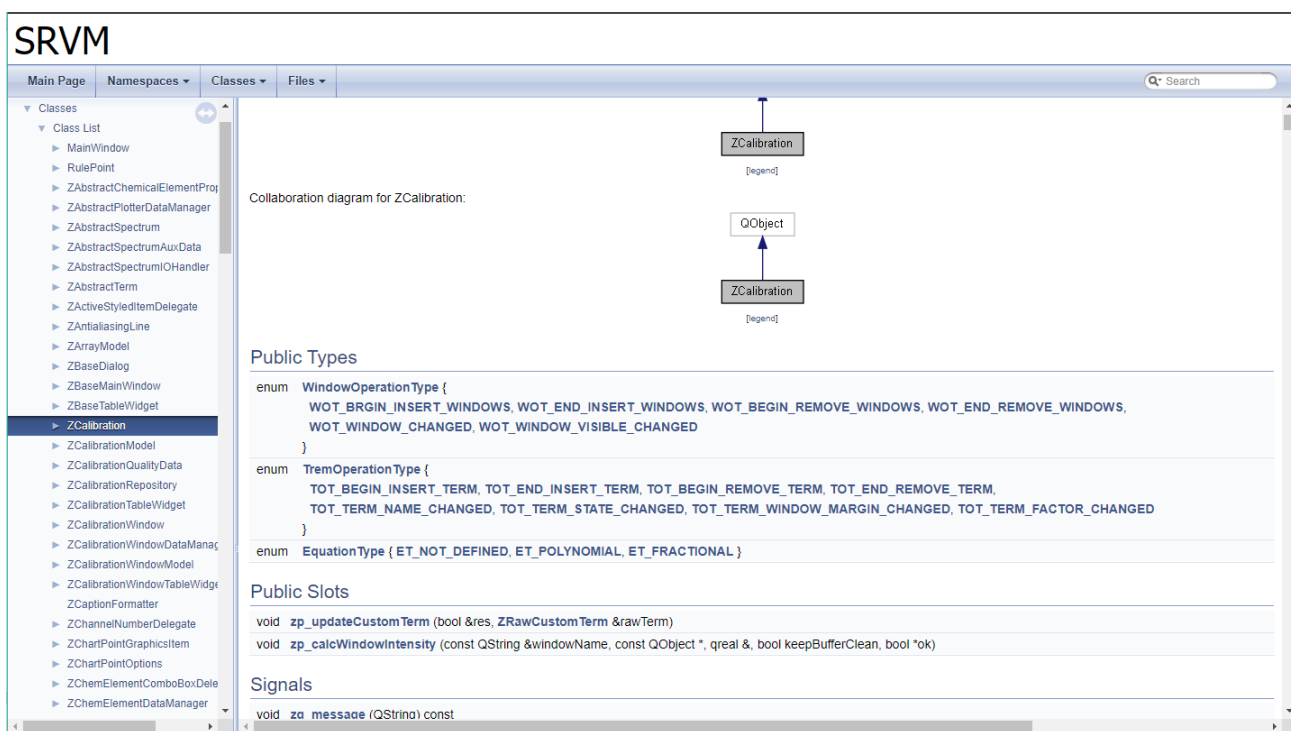


Рисунок 24 - Документация проекта SRVM, сгенерированная Doxygen

Для корректной работы программного продукта все модули и динамические библиотеки должны быть должным образом установлены на компьютере. Для дальнейшей деинсталляции программного продукта и осуществления вызова из контекстного меню требуется произвести соответствующие записи в реестр компьютера. Все эти операции реализованы в инсталляционном пакете программного продукта созданного при помощи системы создания инсталляционных пакетов Inno Setup (Рисунок 25).

Весь инсталляционный пакет со всеми дополнительными модулями и динамическими библиотеками заключён в одном исполняемом файле «SRVMSetup-20.12.03.r.exe» размером 15732 килобайт.

```
SRVM-20.12.03.r.iss - Inno Setup Compiler 6.1.2
File Edit View Build Run Tools Help
; !!! GENERATE NEW GUI !!!
[Setup]
; NOTE: The value of AppId uniquely identifies this application.
; Do not use the same AppId value in installers for other applications.
; (To generate a new GUID, click Tools | Generate GUID inside the IDE.)
AppCopyright = Copyright © TechnoAnalyt Ltd. 2017-2020. All rights reserved.
AppId={2BA890BF-DAA5-4578-B64D-9306D604B37B}
AppName={#MyAppWindowName}
AppVersion={#MyAppVersion}
AppVerName={#MyAppExeBaseName}
AppPublisher={#MyAppPublisher}
AppPublisherURL={#MyAppURL}
AppSupportURL={#MyAppURL}
AppUpdatesURL={#MyAppURL}
CreateUninstallRegKey=yes
DefaultDirName={pf}\{#MyAppShortPublisher}\{#MyAppExeBaseName}
DefaultGroupName={#MyAppWindowName}
OutputBaseFilename={#MyAppName}Setup-{#MyAppFullVersion}
Compression=lzma
SolidCompression=yes

WizardImageFile={#InnoSetupFolder}\WizModernImage-IS.bmp
WizardSmallImageFile={#InnoSetupFolder}\WizModernSmallImage-IS.bmp

[Languages]
Name: "english"; MessagesFile: "compiler:Default.isl"; LicenseFile: "{#DocFolder}\licenseEn.txt"
Name: "russian"; MessagesFile: "compiler:Languages\Russian.isl"; LicenseFile: "{#DocFolder}\licenseRu.txt"
;InfoBeforeFile={#SourceFolder}\before.txt
;InfoAfterFile={#SourceFolder}\after.txt

[CustomMessages]
;QtPage
english.QtPageCaption=Select Destination Location of Qt({#QtVersion}) Library
english.QtPageDescription=Where should Qt({#QtVersion}) Library be installed?
english.QtPageSubCaption1=Setup will install Qt({#QtVersion}) Library into the following folder.
english.QtPageSubCaption2=To continue, click «Next». If you would like to select a different folder, click «Brows
;english.QtPageNewFolderName = PNF

russian.QtPageCaption=Выбор папки установки библиотек Qt({#QtVersion})
russian.QtPageDescription=В какую папку Вы хотите установить библиотеки Qt({#QtVersion})?
russian.QtPageSubCaption1=Программа установит библиотеки Qt({#QtVersion}) в следующую папку.
russian.QtPageSubCaption2=Нажмите «Далее», чтобы продолжить. Если Вы хотите выбрать другую папку, нажмите «Обзор»
;russian.QtPageNewFolderName = Нажмите «Папку» чтобы продолжить. Если вы хотите выбрать другую папку, нажмите «
```

Рисунок 25 - Разработка инсталлятора в Inno Setup

4. Тестирование и результаты

В результате проведенных многочисленных вычислительных экспериментов с градуировочными наборами спектров были получены градуировки для различных химических элементов. На Рисунках 26-29 представлены их диаграммы рассеяния.

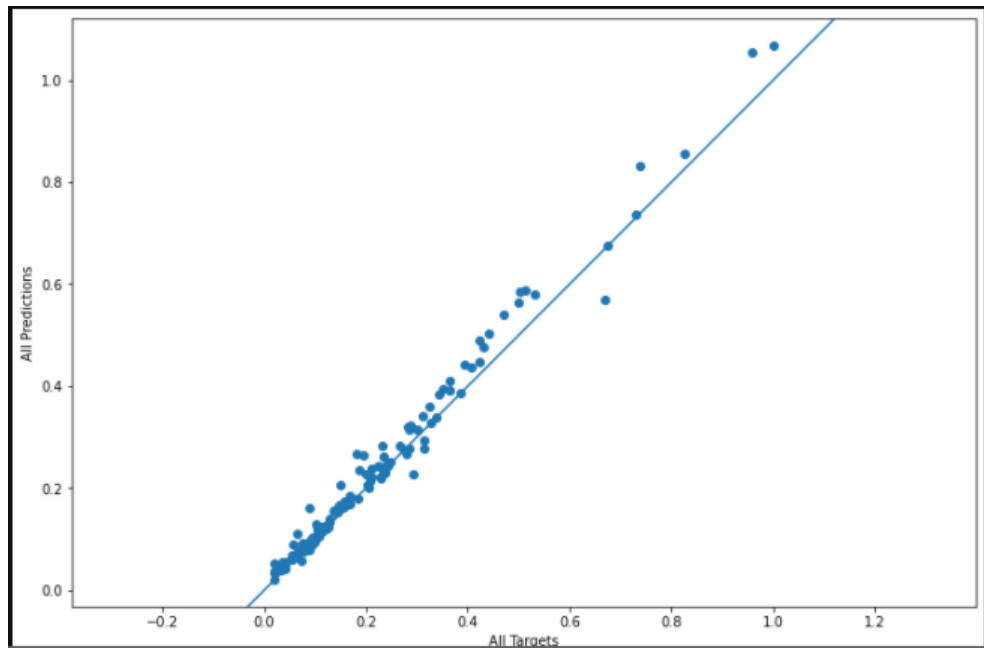


Рисунок 26 - Диаграмма рассеяния градуировки для железа (Fe), $R^2=0,969$

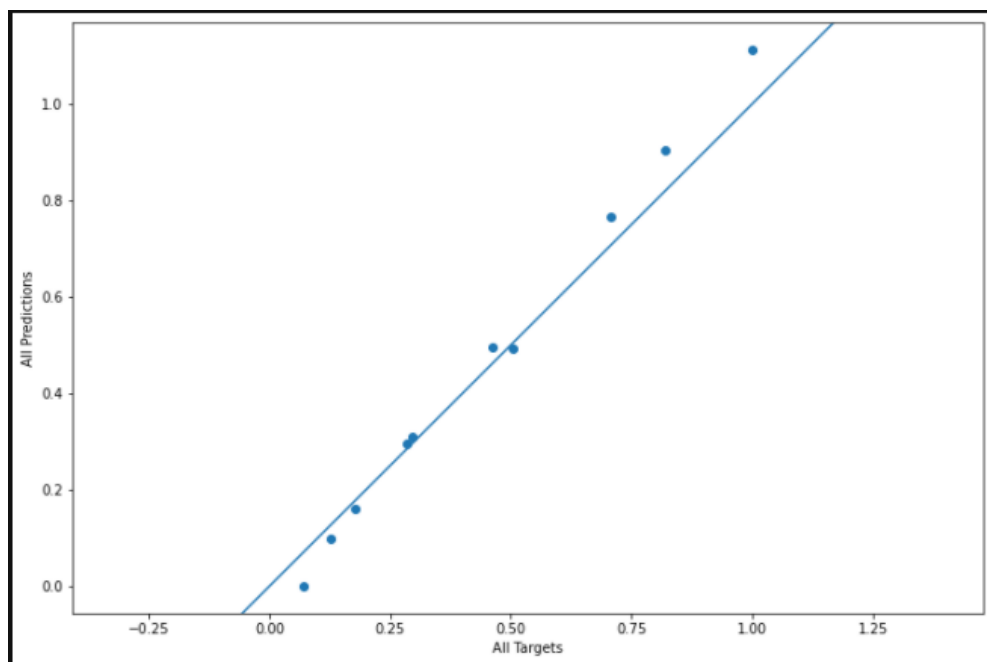


Рисунок 27 - Диаграмма рассеяния градуировки для золота (Au), $R^2=0,936$

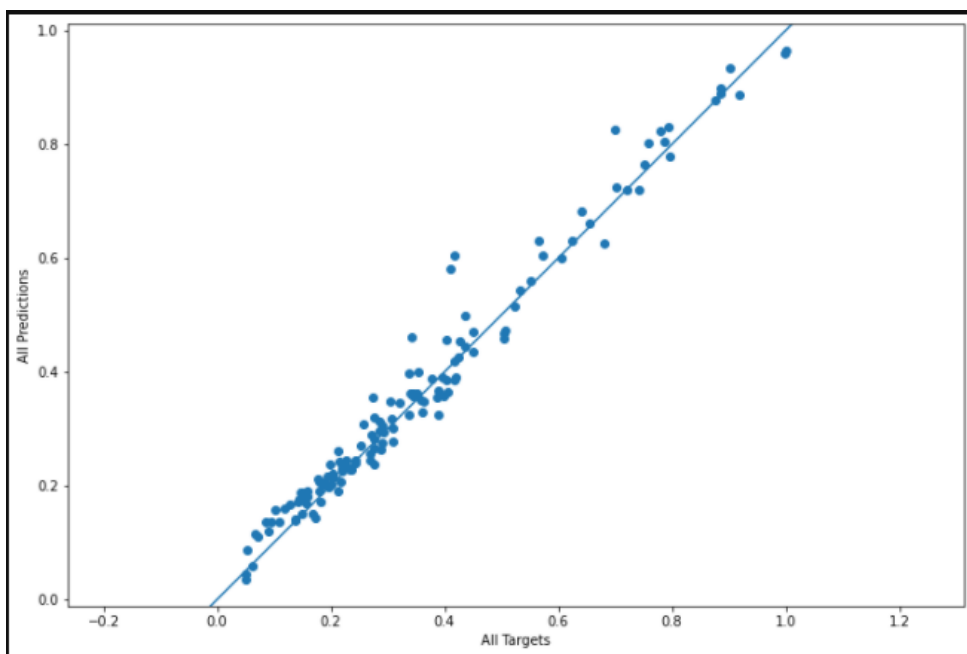


Рисунок 28 - Диаграмма рассеяния градуировки для оксида кремния (SiO₂),

$$R^2=0,941$$

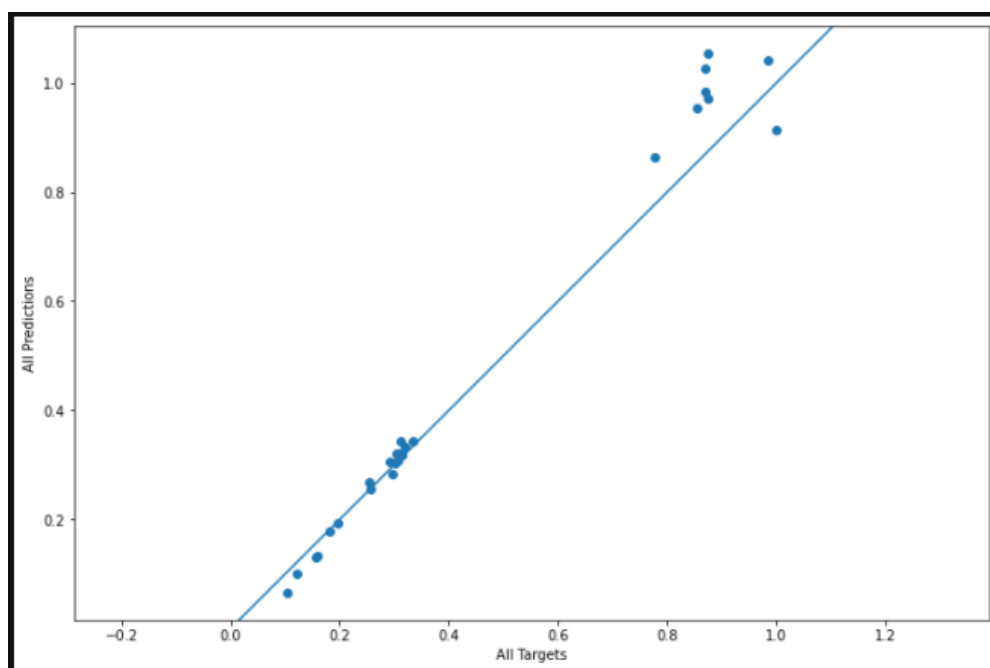


Рисунок 29 - Диаграмма рассеяния градуировки для нитратов (NO₃), $R^2=0,922$

Коэффициент детерминации R^2 рассчитанный на тестовых выборках спектров был не ниже 0,9. Время обучения моделей зависело от размера обучающей выборки и количества эпох обучения, но не превышало 10 минут.

Большинство моделей обучение проходило стабильно, эффекты переобучения не проявлялись (Рисунок 30).

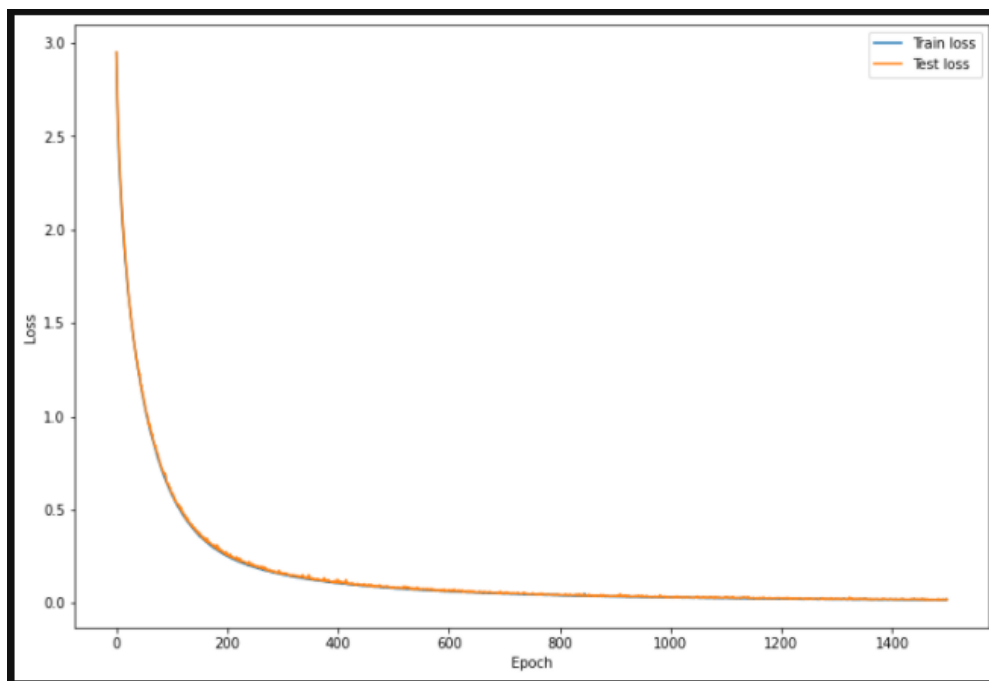


Рисунок 30 – График ошибки на обучающей и тестовой выборках для оксида кремния SiO

Некоторая нестабильность во время обучения наблюдалась при создании градуировок для определения концентраций золота (Рисунок 31) и нитратов (Рисунок 32).

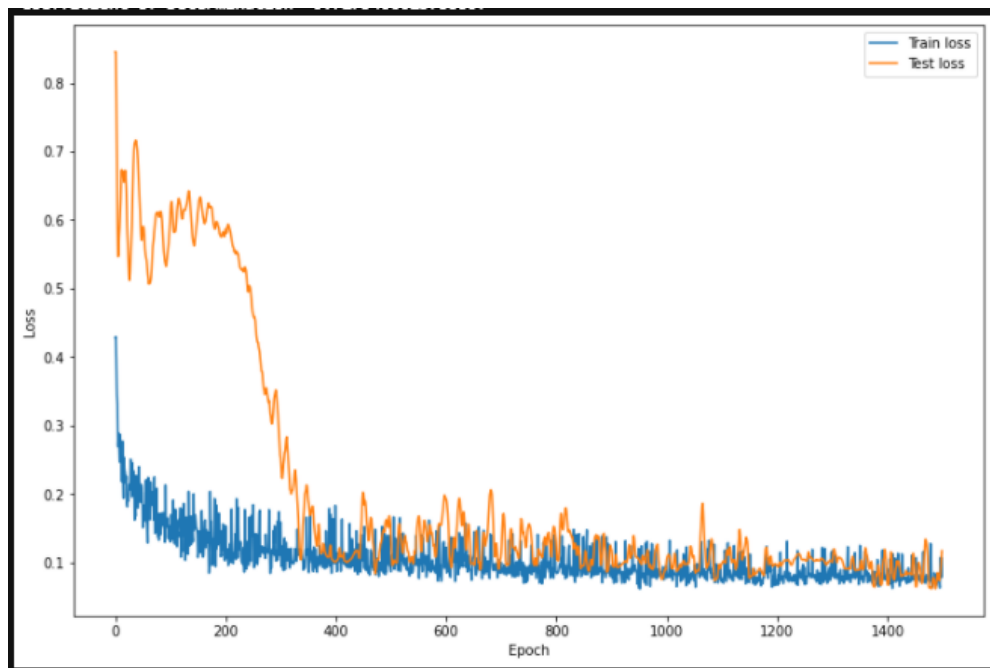


Рисунок 31 - График ошибки на обучающей и тестовой выборках для золота (Au)

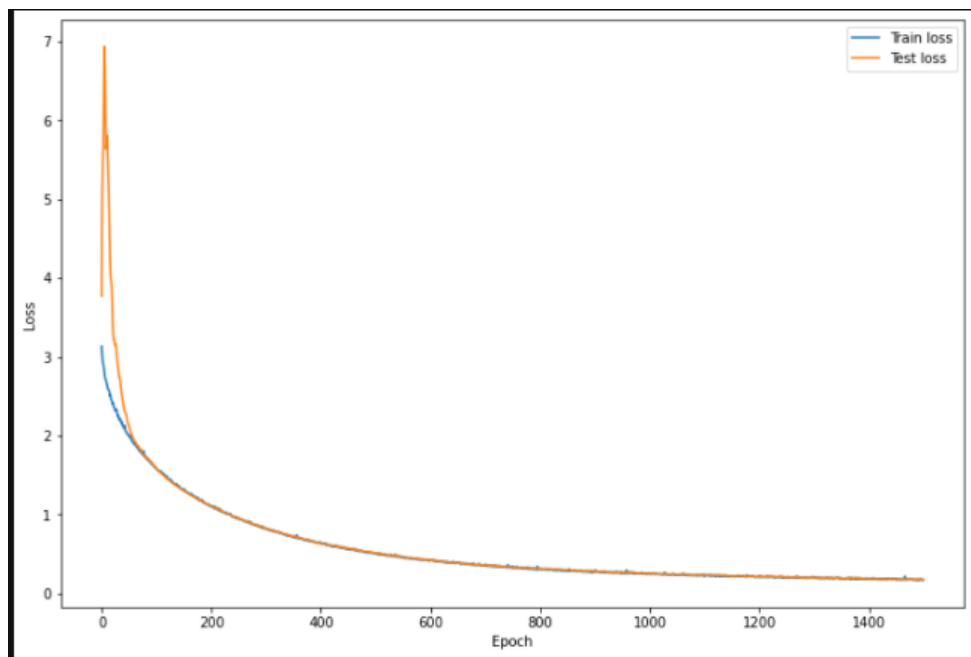


Рисунок 32 - График ошибки на обучающей и тестовой выборках для нитратов (NO₃)

Для проведения тестов и определения возможностей метода, а также с целью демонстрации результатов заинтересованным лицам, были выбраны задачи, либо с трудом решаемые, либо вообще не решаемые традиционными методами. К таким задачам относятся определение концентрации нитратов в технологических растворах обогащения урана и определение концентрации золота, в присутствии в исследуемом материале свинца в высоких концентрациях. В обоих случаях при помощи разработанного программного продукта удалось создать градуировки с приемлемым показателем качества R^2 .

Заключение

В ходе работы над настоящей магистерской диссертацией был разработан метод создания градуировки спектрометра при помощи алгоритмов глубокого обучения нейронной сети без какой-либо предварительной обработки спектров. Признаками, описывающими спектры, являлись значения интенсивностей излучения в каналах спектра. Преодолены связанные с таким подходом трудности, обусловленные мультиколлинеарностью признаков и небольшим размером обучающих выборок. Коэффициент детерминации R^2 , рассчитанный на тестовых выборках спектров был не ниже 0,9. В качестве функции потерь функция абсолютной ошибки была несколько лучше чем среднеквадратическая функция. Полученные результаты позволяют утверждать, что созданные с помощью алгоритмов глубоких нейронных сетей градуировки не уступают по качеству таковым, созданным высококвалифицированными специалистами. Применение разработанных методов позволяет получить значительный экономический эффект.

Метод был реализован в виде тестового модуля на языке Python в среде Jupyter Lab и после испытаний интегрирован в программное обеспечение спектрометра СРВ-1 В. Разработанный модуль был испытан на реальных градуировочных спектрах. Градуировки, полученные при помощи модуля, обладают стабильно высоким качеством. Таким образом, разработанный метод позволяет получать градуировки напрямую из спектров. Работа с программным модулем не требует высококвалифицированного персонала. Результаты вычислительного эксперимента были обсуждены со специалистами компании ТОО «ТехноАналит», производящей рентгенофлуоресцентные спектрометры и специалистами предприятий цветной металлургии и уранодобывающих предприятий. Метод получил высокую оценку. Разработана программа дальнейших испытаний программного модуля и его дальнейшей интеграции в программное обеспечение спектрометров СРВ, производимых ТОО «ТехноАналит».

5. Концепция стартап-проекта

5.1 Проблема конечного потребителя

Использование спектрометров для качественного и количественного элементного анализа во всех индустриальных областях постоянно расширяется благодаря таким их неоспоримым достоинствам по сравнению с традиционными химическими методами как

- высокая скорость получения результатов
- одновременное определение концентраций нескольких элементов
- возможность анализировать вещества в любом агрегатном состоянии и широком диапазоне концентраций от 100 до 10^{-6} %
- исследуемый материал не разрушается
- технологичность (нет необходимости использования реактивов, химической посуды, дополнительного оборудования и т.д.)
- низкая удельная стоимость анализов
- химическая безопасность
- возможность встраивать спектрометры непосредственно в технологический процесс в качестве датчика
- невысокие требования к квалификации персонала, непосредственно проводящего анализ

Тем не менее, существует ряд проблем, связанных с использованием спектрометров. Одной из них является необходимость градуировки спектрометра.

Градуировка представляет собой программный объект, в котором содержатся:

значения настроек спектрометра, при которых должен быть снят спектр (экспозиция в секундах, коэффициент усиления детектора и т.д.),

математическая модель - градуировочное уравнение, по которому определяются концентрации.

Для разработки градуировки требуется специалисты высокой квалификации. Для создания некоторых «сложных» градуировок могут требоваться недели работы дорогостоящих высококлассных специалистов. Кроме того, в реальных условиях очень часто требуется скорректировать имеющиеся градуировки, что тоже требует участия специалистов. Как правило, градуировку спектрометра производят специалисты непосредственно фирм-производителей спектрометра либо их авторизованные представители.

В процессе разработки градуировки специалист вручную выбирает в спектрах базовый пик, определяет наличие влияющих пиков, принимает решение о включении их в математическую модель градуировки, подбирает варианты нормировки спектров, конструирует искусственные независимые признаки. Правильный выбор этих параметров критически влияет качество градуировки и, как следствие, на точность результатов анализа.

В представленном проекте разработан и реализован метод создания градуировочной модели при помощи технологий машинного обучения, вообще не требующий участия квалифицированных специалистов.

Для создания градуировки предложенным методом требуется только загрузить в программное обеспечение файлы спектров материалов с известными химическими концентрациями. Далее, программное обеспечение само создает градуировку.

На тестовой версии разработанной программы были получены градуировочные модели различной сложности, включая градуировку, над созданием которой «классическим» методом высококвалифицированный специалист работал несколько недель. Качество градуировочных моделей было

не хуже, а в некоторых случаях лучше, чем у моделей, созданных «классическим» методом. Время, за которое нейросеть создает модель составляет несколько минут, что является вполне приемлемым показателем.

5.2 Способы защиты интеллектуальной собственности

Основное ноу-хау метода заключается в методах обучения модели и способе создания архитектуры нейросети. Всё это может быть «спрятано» в коде программного обеспечения. Поэтому реверс-инжиниринговый анализ архитектуры ML-структур для раскрытия ноу-хау с целью повторения – малоэффективен.

5.3 Объем и емкость рынка

Результаты исследования рынка спектрометров распространяются платно, но по открытым информационным сегментам прогнозируется, что «...объем рынка масс-спектрометрии вырастет примерно с 4,1 миллиарда долларов США в 2020 году до 5,6 миллиарда долларов к 2025 году при среднегодовом темпе роста 6,5 %. Увеличение расходов на фармацевтические исследования и разработки по всему миру, правительственные постановления о безопасности лекарственных средств, растущее внимание к качеству продуктов питания, рост добычи нефти и сланцевого газа, растущие инициативы правительств по контролю за загрязнением окружающей среды и экологическим испытаниям – являются драйверами роста для рынка спектрометрии в прогнозный период» (по данным консалтинговой компании MarketsandMarkets™ Research Private Ltd.).

При этом три четверти рынка принадлежит Европе и Северной Америке. Цены на лабораторные спектрометры производства стран бывшего СССР составляют в среднем около 60 тысяч долларов, спектрометры ведущих производителей такого же класса - от 130 тысяч долларов.

По этим данным можно приблизительно оценить количество выпускаемых спектрометров – около 30 тысяч в год.

Доля стоимости программного обеспечения в цене спектрометра составляет 10-15 %. Это приблизительно 6-10 тысяч долларов для наших спектрометров и 10-20 тысяч для спектрометров европейских и американских производителей. В общем объеме рынка цена составляет от 450 до 670 миллионов долларов.

Цена градуировки спектрометра на один продукт составляет около 650 долларов для наших спектрометров и около 5-10 тысяч долларов для европейских и американских спектрометров.

Количество градуировок, созданных на один спектрометр за время его эксплуатации, можно оценить от одного десятка до нескольких. Таким образом, расходы на услуги по созданию градуировок могут составлять до десятков тысяч долларов, что уже сопоставимо со стоимостью самого спектрометра.

Таким образом предприятия, использующие спектрометры экономически заинтересованы в программном обеспечении, позволяющем делать качественные градуировки силами обычных лаборантов и исключить соответствующие расходы.

5.4 Современное состояние и перспективы отрасли

Качественное и количественное определение химического состава материалов является одной из фундаментальных основ многих технологических процессов в самых различных индустриальных областях таких как металлургия, фармацевтика, медицина, пищевая промышленность, добыча и переработка полезных ископаемых, химические производства, мониторинг окружающей среды, ювелирное производство, таможенное дело и т.д. Классические методы химического анализа, заключающиеся в проведении ряда химических преобразований проб материалов с целью выделения из них чистого элемента и определения таким образом химической концентрации обладают рядом

недостатков, таких как высокая стоимость, длительное время анализа, технологическая сложность, высокие требования к квалификации персонала и т.д. Поэтому альтернатива в виде анализа спектров получила своё развитие. По мере усовершенствования технологии анализа спектров таких как использования компьютеров, переход на новые типы детекторов, не требующих криогенных температур, общее удешевление электронных и механических компонентов, использование спектрометров расширяется. На данном этапе происходит дальнейшее расширение использования спектрометров в технологических процессах. Спектрометры встраиваются в технологические процессы уже в качестве датчиков, позволяющих осуществлять автоматическое управление и регулировки. Таким образом, можно заключить, что в обозримой перспективе объем рынка спектрометров будет увеличиваться.

5.5 Конкурентные преимущества создаваемого продукта

Анализ продукции ведущих фирм, выпускающих спектрометры, показывает, что их программное обеспечение использует только «классические» методы. Как уже указывалось эти методы требуют работы высококвалифицированных специалистов, что значительно повышает операционные расходы на использования спектрометров. Экспериментальное использование методов машинного обучения для обработки спектров описаны в некоторых научных статьях. Тем не менее, повторение описанных в статьях методов применительно к рентгеновским спектрам дали неудовлетворительные результаты.

Тестирование предложенного продукта выявили следующие конкурентные преимущества:

- оптимизация операционных издержек эксплуатации спектрометров,
- стабильно высокое качество градуировок,
- низкая чувствительность к шумам и сдвигам в спектрах.

5.6 Целевые сегменты потребителей создаваемого продукта

Развивать бизнес на основе предложенного продукта можно параллельно по моделям B2B и B2C.

5.6.1 Модель B2B

Потребителями, очевидно, являются компании, производящие спектрометры. Их основные потребности и мотивации – повышение конкурентоспособности своей продукции и спроса на неё.

Контракты с такими потребителями заключаются на сотни копий единоразово со значительными оптовыми скидками. Возможны также эксклюзивные контракты. Кроме того, такие потребители более лояльны и не склонны без особых причин менять поставщиков.

При этом может возникнуть конфликт интересов, так как создание градуировок – это тоже хороший бизнес для компаний и внедрение подобных продуктов может снизить доходы от него. Компании будут заинтересованы в продукте, если выгоды от его внедрения будут перекрывать связанные с ним издержки.

Так же для заключения таких сделок может потребоваться работа специалистов в области права, что повлечет дополнительные расходы.

5.6.2 Модель B2C

Потребители – предприятия, использующие спектрометры в своей деятельности. Их основные потребности и мотивации – снижение расходов на эксплуатацию спектрометров при сохранении соответствующих качественных показателей. Идеальный спектрометр для таких потребителей – тот который вообще не требует градуировок и работающий по принципу: в него загрузили пробы, он выдал концентрацию. Предлагаемый продукт приближает спектрометр к этому идеалу. Основные проблемы при такой модели – это сопряжение программного продукта со спектрометрами различных производителей. При этом со стороны производителей может возникнуть

противодействие, так как компании заинтересованы в использовании именно их программного обеспечения с их спектрометрами. Тем не менее, случаи использования стороннего программного обеспечения встречаются.

5.7 Цена программного продукта

Так как спектрометры поставляются укомплектованными программным обеспечением, конечные потребители должны быть экономически заинтересованы в приобретении нашего программного продукта. Экономическая мотивация для потребителей, непосредственно использующих спектрометры является снижение затрат на градуировки. Исходя из цены одной градуировки в 40000 рублей и экономии при десяти градуировках устанавливаем цену для этого сегмента потребителей 400000 рублей.

Производители спектрометров заключают контракты на большое количество копий одновременно, поэтому разумно предоставить таким потребителям значительную скидку. Устанавливаем цену для этого сегмента потребителей 200000 рублей.

5.8 Финансирование и продвижение программного продукта

Так как продукт рассчитан на специализированную, узкопрофессиональную аудиторию, его рекламирование и продвижение наиболее эффективно на специальных выставках, конференциях и специализированных интернет и печатных изданиях. Также возможно использовать таргетированную рекламу в поисковых системах. Необходимо также создания интернет-сайта продукта. Все эти способы рекламы и продвижения не потребуют больших финансовых затрат.

Для финансирования проекта на первой стадии до доведения программного продукта до версии RTM планируется рассмотреть несколько вариантов.

- Инвестиции заказчиков. Существующей тестовой версии программного продукта достаточно для того, чтобы показать его результаты и возможности представителям фирм-производителей спектрометров, таким, как ТОО «ТехноАналит», АО "ЭЛЕРАН", НПК «Элватех», «Shimadzu», «Bruker», «PANalytical», «Thermo Scientific», «Oxford Instruments Industrial Analysis» и др. Возможно создание дочернего предприятия фирмы производителя.
- Бизнес-ангелы. Для привлечения бизнес-ангелов можно воспользоваться платформами Funded.com, Angel Capital Association, Angel Investment Network.
- Профессиональные соцсети. Инвесторов, знающих и понимающих бизнес, связанный со спектрометрами, можно найти на таких платформах как LinkedIn.

5.9 Бизнес-модель проекта

Для построения нового бизнеса на основе продукта потребуется решить две основные задачи:

- Доведение программного продукта до версии RTM (готов для выхода на рынок)
- Привлечение потребителей продукта

5.9.1 Команда проекта

Для разработки программного продукта и его дальнейшей поддержки потребуется команда специалистов в следующем минимальном ролевом составе: менеджер проекта и архитектор C/C++ уровня senior, два программиста C/C++ уровня middle. Для привлечения потребителей и продвижения программного продукта потребуется один маркетолог.

По данным рекрутингового сайта HeadHunter (hh.ru) заработная плата разработчика C/C++ уровня middle начинается от 70 000 рублей. Заработная

плата разработчика C/C++ уровня senior начинается от 80 000 рублей. Заработные платы маркетологов начинаются от 35 000 рублей. Так как стартап проект на начальном этапе, как правило, ограничен в финансовых ресурсах, заработная плата сотрудникам возможна с опционом. В этом случае зарплата на начальном этапе – минимальна. В то же время у сотрудников появляется заинтересованность в конечном результате.

Для ведения бухгалтерии на начальном этапе можно воспользоваться услугами бухгалтерских бюро.

Итого, команда на начальном этапе составляет 4 человека. Дальнейшие оценки инвестиций рассчитывались исходя из этого количества сотрудников.

5.9.2 Первоначальные инвестиции

Для работы сотрудников требуется приобретение рабочих мест. Расчет стоимости одного рабочего места представлен в Таблицах 5, 6 (цены взяты с сайта www.komus.ru и www.dns-shop.ru)

Таблица 5 - Расчет рабочего места программиста

Наименование	Количество	Цена (руб.)
Письменный стол «Арго» эргономичный	1 шт.	4 485
Кресло офисное EasyChair Comfort GTP	1 шт.	3 298
Компьютер для работы с нейронными сетями (бюджетная сборка)	1 шт.	129 130
Лампа настольная	1 шт.	1 690
Итого		138 603

Таблица 6 - Расчет рабочего места маркетолога

Наименование	Количество	Цена (руб.)
--------------	------------	-------------

Письменный стол «Арго» эргономичный	1 шт.	44 85
Кресло офисное EasyChair Comfort GTP	1 шт.	3 298
Компьютер для работы с нейронными сетями (бюджетная сборка)	1 шт.	25 210
Лампа настольная	1 шт.	1 690
Итого		34 683

Производим расчет одноразовых затрат для старта проекта (Таблица 7) расчет затрат на содержание предприятия (Таблица 8). Затраты на заработную плату производим исходя из минимальных уровней зарплат специалистов (Таблица 9).

Таблица 7 - Расчет одноразовых затрат для старта проекта

Наименование	Количество	Цена (руб.)	Сумма (руб.)
Рабочее место программиста	3 шт.	138603	415809
Рабочее место маркетолога	1 шт.	34683	34683
Регистрация ООО	1 шт.	4000	4000
Создание интернет-сайта	1 шт.	25000	25000
Итого			479492

Таблица 8 - Расчет затрат для содержания предприятия

Наименование	Сумма за месяц (руб.)	Сумма за год (руб.)
Аренда офиса 30 кв. м	18000	216000
Затраты на коммуникацию (Интернет в офис 100 Мбит/с для организаций)	10000	120000
Накладные расходы на содержание офиса, включая обслуживание расчетного счета	10000	120000
Командировки	20000	240000
Услуги бухгалтерского бюро	2700	32400
Итого	60700	728400

Таблица 9 - Расчет затрат на заработную плату

Наименование	Количество (чел)	Сумма за месяц (руб.)	Сумма с выплатами (руб.)	Сумма за год (руб.)
Менеджер проекта + архитектор С/С++ уровня senior	1	80000	104160	1249920
Программист С/С++ уровня middle	2	70000	182280	2187360
Маркетолог	1	35000	45570	546840
Итого			332010	3984120

5.9.3 Амортизация оборудования

Приобретенное оборудование для рабочих мест относится ко второй амортизационной группе со сроком полезного использования до 3 лет или 36 месяцев. Стоимость оборудования:

Таблица 10 - Стоимость оборудования

Наименование	Количество	Цена (руб.)	Сумма (руб.)
Рабочее место программиста	3 шт.	138603	415809
Рабочее место маркетолога	1 шт.	34683	34683
Итого			450492

Норма амортизации оборудования:

$$\text{Норма амортизации} = \frac{100\%}{\text{Срок полезного использования}} = \frac{100}{36} = 2.8\%$$

Ежемесячная сумма амортизации оборудования составит:

$$450492 * 2.8\% = 12614 \text{ руб.}$$

5.9.4 Годовые затраты проекта

Итого, годовые затраты проекта составляют:

Затраты на содержание предприятия (Таблица 4) + затраты на заработную плату (Таблица 5) + Сумма амортизации оборудования =

$$728400 + 3984120 + 12614 * 12 = 4863888 \text{ руб.}$$

5.9.5 Разработка программного продукта

Для разработки программного продукта требуется выполнить следующие работы:

- сбор, документирование и анализ требований;
- проектирование, разработка архитектуры, разработка тестовых модулей;
- реализация и тестирование программного продукта;
- интеграционное тестирование программного продукта;
- создание документации и инструкции.

Для сбора, документирования и анализа требований выделяется обычно до 3 недель. Дальнейшие сроки зависят от объема работ, определенного на первом этапе. По опыту, в подобных проектах этап проектирования и разработки архитектуры и этап реализации и тестирования могут частично перекрываться по времени и производиться по гибкой методологии итерациями. Приблизительный срок выполнения этих работ – 20 недель.

На интеграционное тестирование программного продукта, устранение недочетов, написание документации и инструкции потребуется до 4 недель. Итого, на разработку программного продукта и доведения его до версии, готовой к выходу на рынок потребуется до 27 недель или 6,3 месяца.

5.9.6 Себестоимость разработки программного продукта

Себестоимость программного продукта представляет собой сумму затрат на разработку и расходов на содержание и эксплуатацию оборудования, используемого при его создании.

За период разработки программного продукта амортизация оборудования составит:

$$12614 * 6.3 = 79468 \text{ руб.}$$

Заработная плата программистов за период разработки программного продукта составит:

$$(104160+182280) * 6.3 = 1804572 \text{ руб.}$$

Затраты на содержание офиса за период разработки программного продукта составит:

$$(18000+10000+10000+20000+2700) * 6.3 = 382410 \text{ руб.}$$

Итого, себестоимость программного продукта составит:

$$79468+1804572 +382410 = 2266450 \text{ руб.}$$

5.10 Оценка эффективности

На первоначальном этапе выбираем упрощенную систему налогообложения (УСН). В УСН доступны два вида налогов: 6% от выручки и 15% от прибыли (то есть доходы минус расходы). Далее будет произведен расчет эффективности и выбран наиболее приемлемый вид налога.

Для оценки экономической эффективности, инвестиционной привлекательности, а также для разработки стратегии ценообразования, установления значений максимальных скидок, важным параметром является точка безубыточности. Точка безубыточности показывает при каком минимальном объеме продаж предприятие перестает быть убыточным и начинает приносить прибыль. Точка безубыточности может быть рассчитана как в деньгах, так и в количестве проданных экземпляров продукта и формируется по смете расходов за месяц с разделением затрат на постоянные и переменные по основным статьям расходов.

Затраты на производство программного продукта, (создание копии программного обеспечения), равны нулю. Переменными затратами являются

только налог на выручку, в случае выбора вида налога 6% от выручки, так как его величина прямо пропорциональна количеству проданных копий продукта. При выборе вида налога 15% от прибыли налог, при расчете точки безубыточности не учитывается, так как до точки безубыточности прибыли нет.

Определяем постоянные затраты (Таблица 11).

Таблица 11 - Постоянные затраты

Наименование	Сумма за месяц (руб.)
Заработная плата	332010
Содержание офиса	73000
Амортизация оборудования	12614
Итого	417624

При налогообложении 6% от выручки (учитывается налог):

Маржинальность = (Выручка – Переменные расходы) / Выручка * 100%

=

$(400000 - 400000 * 0.06) / 400000 * 100 = 94\%$

Точка безубыточности (деньги):

ТБУ (деньги) = Постоянные расходы / Маржинальность * 100% =

$417624 / 94 * 100 = 444281$ руб.

ТБУ (продукт) = ТБУ (деньги) / Выручка =

$444281 / 400000 = 1.11$

При налогообложении 15% от прибыли:

Маржинальность = 100%

Точка безубыточности (деньги):

ТБУ (деньги) = Постоянные расходы / Маржинальность * 100% =

417624 / 100 * 100 = 417624 руб.

ТБУ (продукт) = ТБУ (деньги) / Выручка =

417624 / 400000 = 1.044

Для определения оптимального вида налога производится расчет ежемесячной прибыли в зависимости от количества продаж программного продукта (Таблица 12).

Таблица 12 - Расчет прибыли для различных видов налогов и цен на программный продукт

Количество продаж в месяц	6% от выручки		15% от прибыли	
	Без скидки	Скидка 50%	Без скидки	Скидка 50%
1	-41624	-229624	-17624	-217624
2	334376	-41624	321195.84	-17624
3	710376	146376	657195.84	153195.84
4	1086376	334376	993195.84	321195.84
5	1462376	522376	1329195.84	489195.84
6	1838376	710376	1665195.84	657195.84
7	2214376	898376	2001195.84	825195.84
8	2590376	1086376	2337195.84	993195.84
9	2966376	1274376	2673195.84	1161195.84
10	3342376	1462376	3009195.84	1329195.84

Из Таблицы 12 видно, что, хотя, при количестве продаж, не выходящих на точку безубыточности, убыток при налоге 15% от прибыли меньше, чем при налоге 6% от выручки, при продаже всего двух копий в месяц без скидки и 4 копий со скидкой 50% налог 6% от выручки выгоднее, чем 15% от прибыли. Исходя из этого выбираем вид налога – 6% от выручки.

К первоначальным вложениям в бизнес относятся одноразовые расходы (Таблица 3) и затраты на разработку программного продукта.

Общая сумма первоначальных вложений:

Первоначальные вложения = себестоимость продукта + одноразовые расходы = 2343940 + 479492 = 2823432 руб.

Расчет срока окупаемости проекта в зависимости от ежемесячных продаж:

Таблица 13 - Расчет срока окупаемости проекта в зависимости от ежемесячных продаж

Количество продаж в месяц	Срок окупаемости (мес.)	
	Без скидки	Скидка 50%
1	-68.2162	-136.432
2	8.491734	16.98347
3	3.997083	7.994166
4	2.613673	5.227347
5	1.941657	3.883313
6	1.544533	3.089066
7	1.282272	2.564544
8	1.096147	2.192293
9	0.957206	1.914411
10	0.849525	1.69905

Из Таблицы 13 видно, что при модели В2С и продаже всего трех копий продукта в месяц срок окупаемости проекта составляет 10 месяцев. При скидке 50% для окупаемости за этот срок нужно продать всего 5 копий продукта.

При реализации продукта по модели В2В заключаются оптовые контракты с фирмами-производителями спектрометров. При заключении контракта с одной небольшой компанией, с объемом производства 200 спектрометров в год, выручка после уплаты налогов составит 37 182 376 руб. Чистая прибыль – 32,170,888 руб.

Рентабельность инвестиций составит

$$ROI = (\text{Суммарный доход} - \text{себестоимость}) / \text{Сумма инвестиций} * 100\% = (37182376 - 5011488) / (479492 + 876000 + 3984120) * 100 = 602.49 \%$$

5.11 Заключение

Предложенный проект является инновационным, конкурентноспособным продуктом, способным решить реальные проблемы потребителей. Использование продукта приносит потребителю ощутимый экономический эффект. С точки зрения инвестиционной привлекательности проект не требует больших финансовых вложений. В то же время имеет высокую рентабельность инвестиций – более 600 %.

6. Социальная ответственность

В настоящее время для качественного и количественного элементного анализа материалов во всех индустриальных областях широко используются рентгенофлуоресцентные спектрометры. Полученные при помощи спектрометра спектры анализируются специальным программным обеспечением при помощи математических моделей, называемых градуировками. Создание качественной градуировки является трудоёмким, длительным процессом, требующим от создающего её специалиста опыта и высокой квалификации.

В настоящей магистерской диссертации разработан и реализован в виде программного продукта метод создания градуировок при помощи алгоритмов глубокого обучения нейронных сетей.

Эксплуатация программного продукта осуществляется в лабораториях химического анализа промышленных предприятий различных отраслей.

Исследовательские работы, проектирование и реализация программного продукта проводились в научно-исследовательской лаборатории ТОО «ТехноАналит», город Усть-Каменогорск. Рабочее место состоит из:

- углового рабочего стола,
- эргономичного поворотного кресла с подголовником, снабженного регулировкой по высоте сиденья и наклона спинки,
- персонального компьютера с монитором размером по диагонали 22 дюйма,
- настольного спектрометра СРВ-1В.

Оконные проёмы расположены слева от рабочего места, помещение оборудовано кондиционером, системой парового отопления, потолочными светодиодными панелями в количестве 9 шт.

Использование программного продукта позволяет повысить производительность труда сотрудников, работающих со спектрометром, значительно сократить расходы на обслуживание, повысить качество

градуировок и, как следствие, точность элементного анализа. Создание градуировок при помощи программного продукта не требует привлечения высококвалифицированных специалистов и может быть осуществлена лаборантами химической лаборатории.

6.1 Правовые и организационные вопросы обеспечения безопасности

6.1.1 Специальные (характерные для проектируемой рабочей зоны) правовые нормы трудового законодательства

Работа в лаборатории относится к второй категории тяжести труда с допустимыми условиями труда. Продолжительность рабочего времени – 40 часов в неделю. В течение одного рабочего дня предусмотрен обеденный перерыв на один час и два пятнадцатиминутных перерыва в первой и второй половине дня.

Работа с использованием спектрометра регулируется СанПиН 2.6.1.3289-15 [24], СанПиН 2.6.1.2523-09 [25] и СП 2.6.1.2612-10 [26]. К работе со спектрометрами допускаются лица не моложе 18 лет, не имеющие медицинских противопоказаний, прошедшие соответствующее инструктаж и обучение по радиационной безопасности. Администрация предприятия должна обеспечить безопасные условия работы, проводить инструктаж по технике безопасности и осуществлять ежеквартальный индивидуальный дозиметрический контроль [24].

Работа на персональном компьютере регулируется санитарными правилами СП 2.2.3670-20 "Санитарно-эпидемиологические требования к условиям труда", раздел XXII [27] и "ТОИ Р-45-084-01. Типовая инструкция по охране труда при работе на персональном компьютере" [28].

В соответствии с ТОИ Р-45-084-01 [28], к работе на персональном компьютере допускаются лица, прошедшие обучение технике безопасности и непосредственный инструктаж на рабочем месте. Виды работ на персональном

компьютере делятся на 3 категории тяжести: А – работа по считыванию с экрана персонального компьютера; Б – работа по вводу информации; В - творческая работа в режиме диалога с компьютером. Работа с использованием спектрометра относится к категории В. Для этой категории работ устанавливается суммарное время работы непосредственной на персональном компьютере не более 6 часов за смену. При рабочей смене длительностью 8 часов устанавливаются перерывы через 2 часа после начала рабочей смены и через 2 часа после обеденного перерыва длительностью по 15 минут каждый. Длительность обеденного перерыва устанавливается не менее 40 минут.

6.1.2 Основные эргономические требования к компоновке рабочей зоны

Рабочее место для работы со спектрометром должно соответствовать ГОСТ 12.2.032-78 «Рабочее место при выполнении работ сидя» [29] и отвечать требованиям "ТОИ Р-45-084-01 [28]. Типовая инструкция по охране труда при работе на персональном компьютере". Основные требования к компоновке, в соответствии с этими документами:

- площадь на одно рабочее место должно составлять не менее 4,5 кв.м.,
- оконные проёмы в помещении должны быть оборудованы устройствами, позволяющими регулировать освещение (жалюзи, козырьки и т.п.) и располагаться так, чтобы свет от них падал сбоку, преимущественно, слева,
- рабочий стол должен иметь пространство для размещения ног высотой не менее 600 мм. и глубиной не менее 450 мм на уровне колен и не менее 650 мм на уровне стоп,
- рабочий стол должен иметь размеры для размещения всего комплекта оборудования и необходимого объёма рабочей документации,

- высота рабочей поверхности стола должна регулироваться в пределах 680 – 800 мм., либо составлять 725 мм, при отсутствии возможности регулировки,
- рабочий стул или кресло должно быть оснащено подъемно - поворотным механизмом и регулируемым по высоте и углам наклона спинки.

Рабочее место должно быть скомпоновано так, чтобы все операции работника выполнялись в пределах зоны досягаемости моторного поля в вертикальной плоскости (Рисунок 33) и в горизонтальной плоскости (Рисунок 34).

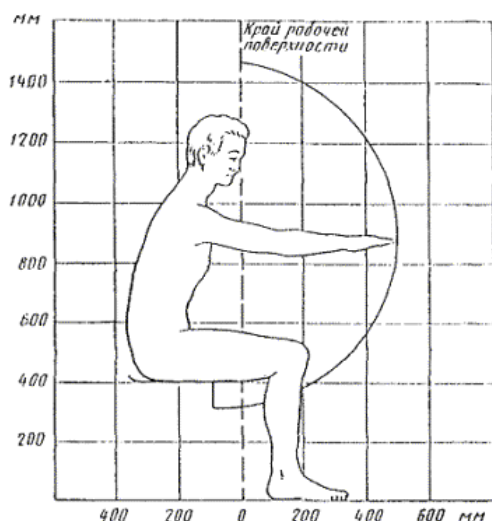


Рисунок 33 - Зона досягаемости моторного поля в вертикальной плоскости

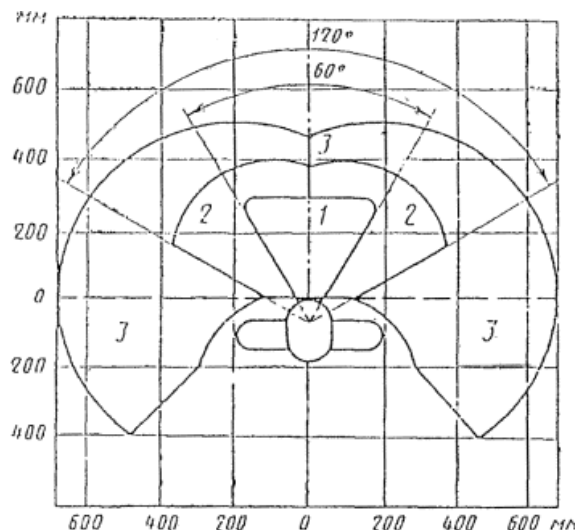


Рисунок 34 - Зона досягаемости моторного поля в горизонтальной плоскости.

1 - зона для размещения наиболее часто используемых предметов

2 - зона для размещения часто используемых предметов

3 - зона для размещения редко используемых предметов

Пример комфортной компоновки рабочего места на угловом столе приведен на Рисунке 35. Все основные органы управления спектрометром находятся в зоне досягаемости моторного поля. Поворотное офисное кресло

обеспечивает дополнительный комфорт и оперативность управления всеми частями спектрометрического комплекса.

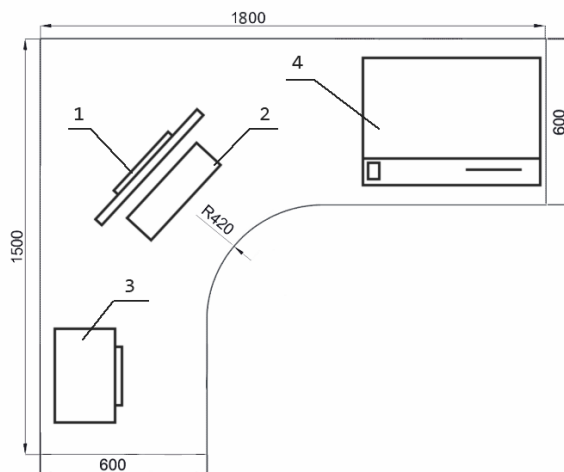


Рисунок 35 - Компонировка рабочего места на ТОО «ТехноАналит». 1-монитор, 2-клавиатура, 3-принтер, 4-спектрометр

Дисплей персонального компьютера должен располагаться на рабочем столе прямо, на удалении от глаз работника на 500 – 600 мм. в границах оптимальной зоны (Рисунок 36), так, чтобы изображение в любой его части читалось без поднятия или поворота головы.

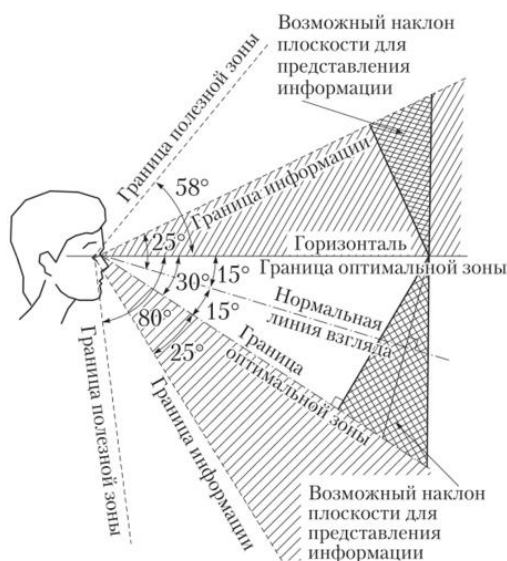


Рисунок 36 - Зоны зрительного наблюдения в вертикальной плоскости

Для комфортной длительной работы важны также технические параметры дисплея, такие как размер, разрешение, тип матрицы. Рекомендуемые параметры:

- диагональ – 21-24 дюйма;
- разрешение – 1280 на 1024;
- тип матрицы – IPS.

6.2 Производственная безопасность

В процессе разработки программного продукта и его эксплуатации на работника воздействует ряд неблагоприятных производственных факторов.

В таблице 14 представлены выявленные в процессе анализа производственной среды вредные и опасные факторы.

Таблица 14 - Вредные и опасные факторы при разработке и эксплуатации программного продукта для обработки рентгеновских спектров

Факторы (ГОСТ 12.0.003-2015)	Этапы работ		Нормативные документы
	Разраб отка	Эксплу атация	
Опасные и вредные производственные факторы, связанные со световой средой	+	+	СП 52.13330.2016 Естественное и искусственное освещение
Опасные и вредные производственные факторы, связанные с аномальными микроклиматическими параметрами воздушной среды на местонахождении работающего	+	+	СанПиН 2.2.4-548-96 Гигиенические требования к микроклимату производственных помещений
Коротковолновое электромагнитное излучение (поток фотонов высоких	+	+	СанПиН 2.6.1.3289-15 Гигиенические требования по обеспечению

<p>энергий) - рентгеновское излучение и гамма-излучение;</p>			<p>радиационной безопасности при обращении с источниками, генерирующими рентгеновское излучение при ускоряющем напряжении до 150 кВ, СанПиН 2.6.1.2523-09 Нормы радиационной безопасности (НРБ-99/2009), СП 2.6.1.2612-10 Основные санитарные правила обеспечения радиационной безопасности (ОСПОРБ 99/2010)</p>
<p>Повышенный уровень и другие неблагоприятные характеристики шума</p>	<p>+</p>	<p>+</p>	<p>СН 2.2.42.1.8.562-96 Санитарные нормы. Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки, ГОСТ 12.1.003-2014 Система стандартов безопасности труда. Шум. Общие требования безопасности</p>
<p>Опасные и вредные производственные факторы, связанные с электрическим током, вызываемым разницей электрических потенциалов, под действие которого попадает работающий.</p>	<p>+</p>	<p>+</p>	<p>ГОСТ 12.1.019-2017 Система стандартов безопасности труда ЭЛЕКТРОБЕЗОПАСНОСТЬ, ТОИ Р-45-084-01 Типовая инструкция по охране труда при работе на персональном компьютере</p>

6.2.1 Анализ опасных и вредных производственных факторов

6.2.1.1 Опасные и вредные производственные факторы, связанные со световой средой

Нормы и требования к освещению на рабочем месте регулируются сводом правил СП 52.13330.2016 [30]. Опасные и вредные производственные факторы, связанные со световой средой, включают в себя:

- отсутствие или недостаток необходимого естественного освещения;
- отсутствие или недостатки необходимого искусственного освещения;
- повышенная яркость света;
- пониженная световая и цветовая контрастность;
- прямая и отраженная блёскость;
- повышенная пульсация светового потока.

Источник возникновения фактора – невозможность организации естественного освещения, отсутствие устройств для его регулировки, неправильное расположение элементов рабочего места в помещении, некачественное устройство систем искусственного освещения.

Плохое освещение вызывает перенапряжение зрения, утомление, головную боль, снижает общую работоспособность персонала.

Лаборатория спектрального анализа, где проводилась разработка программного продукта относится к типу «Лаборатории научно-технические (кроме медицинских учреждений): термические, физические, спектрографические, стилометрические, фотометрические, микроскопные, рентгеновские, рентгено-структурного анализа, механические, радиоизмерительные, электронных устройств» [30]. Освещенность рабочих поверхностей для таких помещений при общем освещении устанавливается не менее 400 лк, при комбинированном – 300 – 500 лк.

Для решения проблемы недостатка освещения производственные помещения оборудуются системами общего и комбинированного – общего и местного искусственного освещения. Использование только местного искусственного освещения рабочего места не допускается. Комбинированное освещение применяется для рабочих мест, на которых осуществляются работы повышенной точности. Работа со спектрометром не относится к категории точных, поэтому возможно применение только общего равномерного освещения.

Для предотвращения бликов на экране монитора рабочее место должно быть расположено так, чтобы источники естественного света были расположены так, чтобы свет от них падал сбоку.

6.2.1.2 Опасные и вредные производственные факторы, связанные с аномальными микроклиматическими параметрами воздушной среды на местонахождении работающего

Нормы и требования к микроклимату регулируются сводом правил СанПин 2.2.4.3359-16 [31]. К параметрам микроклимата в производственных помещениях относятся:

- температура воздуха,
- температура поверхностей,
- относительная влажность воздуха,
- скорость движения воздуха,
- интенсивность теплового облучения.

Нарушения параметров микроклимата влекут за собой дискомфорт, снижение производительности труда, простудные заболевания, повышение кровяного давления, сердечно-сосудистые заболевания.

Параметры микроклимата должны быть такими, чтобы обеспечить оптимальный тепловой баланс работника с окружающей средой. Параметры выбираются исходя из классификации тяжести проводимых в помещении работ

и времени года. Работа со спектрометром выполняется сидя, не требует физического напряжения и по показателям напряженности трудового процесса относится к категории оптимальная (легкая физическая нагрузка) [32]. Для таких помещений устанавливаются следующие оптимальные параметры микроклимата (Таблица 15) [33].

Таблица 15 - Оптимальные параметры микроклимата

Период	Температура, °С	Относительная влажность воздуха, %	Скорость движения воздуха, не более, м/с
Холодный	22-24	40-60	0,1
Тёплый	23-25	40-60	0,1

Нагревающиеся поверхности на спектрометре отсутствуют, как и источники теплового облучения, поэтому соответствующие параметры не рассматриваются.

Для поддержания параметров микроклимата в надлежащих пределах в помещении устанавливаются системы парового отопления с возможностью их регулировки и системы кондиционирования воздуха с мощностью, соответствующей объему производственного помещения.

6.2.1.3 Коротковолновое электромагнитное излучение (поток фотонов высоких энергий) - рентгеновское излучение и гамма-излучение

Требования и нормы к коротковолновому электромагнитному излучению регулируются СанПиН 2.6.1.3289-15 [24], СанПиН 2.6.1.2523-09 [25], СП 2.6.1.2612-10 [26].

Источником коротковолнового электромагнитного излучения является рентгеновская трубка спектрометра.

Превышение интенсивности и энергии коротковолнового электромагнитного излучения может привести к ожогам, нарушению работы внутренних органов, лучевой болезни и смертельному исходу.

Персонал, допущенный к работе со спектрометром, относится к группе А по радиационной опасности. Пределы допустимых доз для группы А приведены в таблице 16 [25].

Таблица 16 - Пределы допустимых доз облучения для персонала группы А

Нормируемые величины	Пределы доз
Эквивалентная доза	20 мЗв в среднем за год за любые последовательные пять лет, но не более 50 мЗв в год
Эквивалентная доза за год	
В хрусталике глаза	150 мЗв
В коже	500 мЗв
В кистях и стопах	500 мЗв

В исправных спектрометрах при соблюдении правил техники безопасности и правил эксплуатации коротковолновое электромагнитное излучение полностью поглощается защитным кожухом и не представляет опасности для персонала. Однако действия, нарушающие эти правила могут привести к облучению ионизирующим излучением. Для своевременного обнаружения наличия ионизирующего излучения рабочее место должно быть оборудовано специальными регистрирующими приборами. Персонал должен

быть проинструктирован и обучен действиям в случае обнаружения ионизирующего излучения. Также должны быть приняты административные меры, исключающие возможность доступа к спектрометру лиц, не имеющих соответствующего допуска.

6.1.2.4 Повышенный уровень и другие неблагоприятные характеристики шума

Требования и нормы к характеристикам шума регулируются сводом правил СН 2.2.42.1.8.562-96 [34] и ГОСТ 12.1.003-2014 [35].

Источник шума – работающие системы охлаждения спектрометра и компьютера, системы кондиционирования воздуха в помещении. Также шум может проникать извне.

Повышенный шум вызывает кумулятивное ухудшение слуха, утомление, головную боль, снижение внимания и реакции. Предельно допустимый уровень звука (нулевой риск потери слуха) равен 80 дБ.

Рабочее место в лаборатории соответствует типу «Творческая деятельность, руко-во-дящая работа с повышенными тре-бо-ваниями, научная деятельность, конструирование и проектирование, программирование, преподавание и обучение, врачебная деятельность. Рабочие места в помещениях дирекции, проектно-конструк-торс-ких бюро, расчетчиков, програм-мис-тов вычислительных машин, в лабораториях для теоретических работ и обработки данных, приема больных в здравпунктах» [34]. Для такого вида работ и типа помещения предельные уровни звукового давления для девяти октавных полос звуковых частот приведен в Таблице 17 [34].

Таблица 17 - Предельно допустимые уровни звукового давления

31,5 Гц	63 Гц	125 Гц	250 Гц	500 Гц	1000 Гц	2000 Гц	4000 Гц	8000 Гц	Уровни звука и эквивален- тные уровни звука (в дБА)
86 дБ	71 дБ	61 дБ	54 дБ	49 дБ	45 дБ	42 дБ	40 дБ	38 дБ	50 дБ

Лаборатория, в которой разрабатывался и используется программный продукт, является помещением с низким уровнем общего шума. Измеренный уровень шума составлял не более 50 дБ.

При значениях шума выше установленных санитарно-гигиенических норм необходимо предпринять меры по его локализации и снижению. Изолировать источник шума акустическими экранами, предусмотреть звукопоглощающие покрытия и облицовки, использовать индивидуальные средства защиты от шума, такие как беруши, противοшумные наушники, антифоны.

6.1.2.5 Опасные и вредные производственные факторы, связанные с электрическим током, вызываемым разницей электрических потенциалов, под действие которого попадает работающий

Требования и нормы к электробезопасности регулируются следующими документами: ГОСТ 12.1.019-2017 [36] и ТОО Р-45-084-01 [28].

Источниками возможного поражения электрическим током на рабочем месте являются все приборы, установки и оборудование подключенные к промышленной или бытовой сети электропитания 220 В.:

- персональный компьютер,

- спектрометр,
- местные и общие осветительные приборы,
- системы кондиционирования воздуха.

Основные причины поражения электрическим током – действия, грубо нарушающие технику безопасности, такие как:

- использование электроприборов со снятыми кожухами или в разобранном виде,
- ремонт электроприборов, не отключенных от электросети,
- неисправность электроприбора, приводящая к появлению электрического напряжения на металлических частях корпуса.

Поражение электрическим током вызывает ожоги, паралич мышц, болевой синдром, поражение внутренних органов. В тяжелых случаях приводит к общему поражению центральной нервной системы, фибрилляции и параличу сердечной мышцы, что может повлечь смертельный исход.

Тяжесть поражения электрическим током зависит от величины тока, протекающего сквозь тело человека, его частоты и длительности этого воздействия. Наибольшую опасность для жизни и здоровья человека представляет ток частота которого находится в пределах от 20 до 100 Гц, опасное значение тока – от 0.001А, смертельное значение тока – от 0.1 А.

Для предотвращения поражения электрическим током персонал обязан строго выполнять предписания техники электробезопасности, не использовать электроприборы в неисправном, разобранном состоянии, на всех электроприборах должно применяться защитное заземление по ГОСТ 12.1.019-2017 [36] Расчет защитного заземления представлен в Приложении В.

6.2.2 Обоснование мероприятий по снижению уровней воздействия опасных и вредных факторов на исследователя (работающего)

Все сотрудники лаборатории обязаны знать и неукоснительно соблюдать все предписания техники безопасности. Персонал должен пройти вводный общий инструктаж по технике безопасности и производственной санитарии и инструктаж на рабочем месте. После инструктажа квалификационная комиссия осуществляет проверку знаний техники безопасности, присваивает каждому сотруднику квалификационную группу и выдает соответствующие удостоверения. Все сотрудники должны пройти медицинское освидетельствование перед приемом на работу и проходить ежегодный медицинский осмотр.

Лаборатория должна быть оборудована приборами общего освещения, обеспечивающими спектр света близкий к дневному и уровень освещения не менее 400 лк. Расположение и мощность приборов должно быть спроектировано и установлено таким образом, чтобы обеспечить равномерность освещения на всей площади лаборатории. Источники естественного света должны располагаться сбоку от рабочего места и иметь возможность регулирования светового потока.

Для контроля микроклиматических параметров в лаборатории должны быть установлены термометр и психрометр. Помещение должно быть оборудовано регулируемыми системами отопления, кондиционирования и вентиляции.

Спектрометр относится к установкам I группы радиационной опасности (низкая опасность) и не требует специальных мер защиты. Всё ионизирующее излучение экранируется металлическим корпусом. Также в спектрометре установлены специальные датчики, прерывающие подачу высокого напряжения на источник излучения при открытии крышки. Тем не менее, для контроля уровня излучения персонал должен быть оснащен специальными регистрирующими устройствами и знать порядок действий при обнаружении излучения, превышающего безопасный уровень.

Замер уровня шума должно производиться не реже одного раза в год. При обнаружении источников шума, превышающих гигиенические нормы, необходимо принять соответствующие меры, связанные с ремонтом, заменой или акустическим изолированием шумящих устройств.

Все используемые в лаборатории электроприборы должны быть исправны и иметь защитное заземление.

6.3 Экологическая безопасность

Жизненный цикл оборудования входящего в состав спектрометра с точки зрения воздействия на экологию можно разделить на три этапа.

1. Производство. На этом этапе происходит потребление электроэнергии, сырья в виде металлов, пластмассы и других материалов, сопутствующих ресурсов в виде водо- и теплоснабжения, горюче-смазочных материалов при решении логистических задач.
2. Эксплуатация. Потребляется электроэнергия.
3. Утилизация. Образуются твердые отходы. В состав оборудования спектрометра входят [37]:
 - трансформаторы, конденсаторы, люминесцентные лампы мониторов – отходы первого класса опасности, для утилизации которых в соответствии с [38], необходима лицензия;
 - электрические компоненты – кабели, платы, пластик и т.д. – отходы третьего класса опасности.

В соответствии с ГОСТ Р 55102-2012 «Ресурсосбережение ОБРАЩЕНИЕ С ОТХОДАМИ» [37] электротехническое оборудование должно быть сдано на специальные предприятия, на которых производится поэтапная разборка и сортировка компонентов для их последующей вторичной переработки или безопасного хранения.

В течение жизненного цикла оборудование спектрометра оказывает влияние на

- атмосферу – энерго- и теплотребление и выбросы, связанные с ним, выбросы, связанные с логистикой;
- гидросферу – потребление водных ресурсов и сбросы в водные источники при производстве материалов, используемых при производстве оборудования спектрометра;
- литосферу – твердые отходы при утилизации оборудования с нарушением предписанных правил.

6.4 Безопасность в чрезвычайных ситуациях

6.4.1 Анализ возможных чрезвычайных ситуаций (ЧС)

При проектировании и эксплуатации программного продукта возможны следующие чрезвычайные ситуации в рабочей зоне:

- техногенные - производственные аварии, пожары;

Наиболее типичным и вероятным видом чрезвычайной ситуации является пожар в рабочей зоне. Наиболее вероятные причины пожара связаны с неисправностью или ненадлежащей эксплуатации электроприборов – короткое замыкание или перегрузки по току, нарушение работником правил эксплуатации.

В соответствии с СП 12.13130.2009 [39] помещение лаборатории относится к категории В1 зона П-Па. Перед тушением необходимо отключить общее электроснабжение. Для тушения можно использовать огнетушители разного типа, но, если отключить электроснабжение не удастся, необходимо применять только порошковые и углекислотные. Требования к эксплуатации огнетушителей приведены в СП 9.13130.2009 [40].

6.4.2 Меры по предупреждению возникновения пожара

Для предупреждения пожаров необходимо предпринять следующие меры:

- организовать инструктаж и практическое обучение персонала по технике пожарной безопасности и пожаротушению;
- оснастить помещение противопожарной сигнализацией, первичными средствами ликвидации возгорания, пригодное для использования на электроустановках;
- помещение лаборатории должно соответствовать всем требованиям пожарной безопасности, иметь аварийные выходы, план эвакуации персонала;
- эксплуатировать только исправное электрооборудование, неукоснительно соблюдать правила эксплуатации, не допускать превышение допустимых нагрузок на электросеть.

6.4.3 Действия в случае возникновения пожара

В случае возникновения пожара необходимо:

- спокойно оценить ситуацию и принять срочные меры по предотвращению распространения огня;
- вызвать пожарных, сообщив точный адрес места возгорания и ФИО, вызывающего;
- отключить общее электроснабжение;
- попытаться потушить пожар при помощи первичных средств пожаротушения;
- если самостоятельно справиться с огнем не удаётся, следует включить сигнал пожарной тревоги и, согласно плану, приступить к эвакуации;
- встретить прибывшую пожарную команду и обеспечить для неё беспрепятственный доступ и пути подъезда к месту пожара.

6.5 Заключение

В настоящей главе рассмотрены вредные и опасные производственные факторы, возникающие при разработке и эксплуатации разработанного программного продукта. Рассмотрены и предложены меры по снижению их воздействия на персонал. Рассчитана система защитного заземления спектрометра. Реализация предложенных мер повышает комфорт и безопасность на рабочем месте, повышает качество и производительность труда. Рассмотрено влияние на экологию, связанное с производством, эксплуатацией и утилизацией оборудования спектрометра. Предложены меры по минимизации этого влияния. Описаны необходимые меры для предотвращения пожара, а также действия в случае его возникновения. Реализация этих мер повышает безопасность персонала, оборудования и имущества.

Список использованных источников

1. А.А.Веригин. Энергодисперсионный рентгенофлуоресцентный анализ. Применение в промышленности. Томск: Издательство ТГУ, 2005. стр. 242.
2. Х.Эрхардт, [ред.]. Рентгенофлуоресцентный анализ. Применение в заводских лабораториях. Москва: Металлургия, 1985. стр. 256.
3. А.В. Бахтиаров. Рентгеноспектральный флуоресцентный анализ в геологии и геохимии. Ленинград: Недра, 1985. стр. 144.
4. Дулов Е.Н., Ивойлов Н.Г. Рентгеноспектральный флуоресцентный анализ. Казань: Издательство Казанского государственного университета, 2008. стр. 50.
5. Н.Ф. Лосев, А.Н. Смагунова. Основы рентгеноспектрального флуоресцентного анализа. Москва: Химия, 1982. стр. 208.
6. Золотарев В.М., Никоноров Н.В., Игнатьев А.И. Современные методы исследования оптических материалов. Учебное пособие. Часть I. Методы исследования состава и структуры материалов. Санкт-Петербург: СПб: НИУ ИТМО, 2013. стр. 266. Т. 1.
7. Р. К. Спиров, А. Н. Никитин. НЕЙРОННЫЕ СЕТИ В СПЕКТРОМЕТРИИ РАДИОАКТИВНЫХ ИЗЛУЧЕНИЙ: СОСТОЯНИЕ ПРОБЛЕМЫ. ЭКОЛОГИЧЕСКИЙ ВЕСТНИК. 01 2016 г., 35.
8. Radtke, A Rakotondrajoa M. Machine learning based quantification of synchrotron radiation-induced x-ray fluorescence measurements—a case study. Machine Learning: Science and Technology. 2020 г., Т. Volume 2, Number 2, 28 December 2020.
9. Conor D Rankin, Thomas J Penfold. A Deep Neural Network for the Rapid Prediction of X-ray Absorption Spectra. The Journal of Physical Chemistry . May 2020 г.
10. Dipendra Jha, Logan Ward, Arindam Paul, Wei-keng Liao, Alok Choudhary, Chris Wolverton & Ankit Agrawal. ElemNet: Deep Learning the Chemistry of Materials from Only Elemental Composition. Scientific Reports. August 2018 г.
11. Dipendra Jha, Vishu Gupta, Logan Ward, Zijiang Yang, Christopher Wolverton, Ian Foster, Wei keng Liao, Alok Choudhary, Ankit Agrawal. Enabling deeper learning on big data for materials informatics applications. Scientific Reports. February 2021 г., 11.
12. G. Drera, C.M. Kropf, L. Sangaletti. Deep neural network for X-ray photoelectron spectroscopy data analysis. Researchgate. [В Интернете] September 2019 г. [Цитировано: 11 02 2021 г.] https://www.researchgate.net/publication/335788471_Deep_neural_network_for_X-ray_photoelectron_spectroscopy_data_analysis.
13. Kunal Ghosh, Annika Stuke, Milica Todorovic', Peter Bjørn Jørgensen, Mikkel N. Schmidt. Deep Learning Spectroscopy: Neural Networks for Molecular

- Excitation Spectra. Advanced Science News WILEY-VCH Verlag GmbH & Co. 2019 г., 6.
14. Ling, Ying Zhang and Chen. A strategy to apply machine learning to small datasets in materials science. npj Computational Materials. May 2018, Vol. 4, 25.
 15. Lu-Ning Lia, Xiang-Feng Liua, Wei-Ming Xua, Jian-Yu Wanga, Rong Shu. A laser-induced breakdown spectroscopy multi-component quantitative analytical method based on a deep convolutional neural network. Elsevier. May 2020 г.
 16. Matthew R. Carbone, Mehmet Topsakal, Deyu Lu, and Shinjae Yoo. Machine-Learning X-ray Absorption Spectra to Quantitative Accuracy. PHYSICAL REVIEW LETTERS. 2020 г., Т. Vol. 124, Iss. 15 — 17 April 2020.
 17. Shimadzu Home. Shimadzu. [В Интернете] [Цитировано: 25 05 2021 г.] <https://www.shimadzu.com/>.
 18. Bruker Home. Bruker. [В Интернете] [Цитировано: 25 05 2021 г.] <https://www.bruker.com/>.
 19. PANalytical Home. PANalytical. [В Интернете] [Цитировано: 25 05 2021 г.] <https://www.malvernpanalytical.com/>.
 20. ThermoFisher Scientific Home. ThermoFisher Scientific. [В Интернете] [Цитировано: 25 05 2021 г.] <https://www.thermofisher.com/>.
 21. Oxford Instruments Industrial Analysis Home. Oxford Instruments Industrial Analysis. [В Интернете] [Цитировано: 25 05 2021 г.] <https://www.oxinst.com/>.
 22. А. А. Сорокин, А. С. Шепотько, С. Ю. Страхов, А. С. Стукалова. ПРИМЕНЕНИЕ ТЕХНОЛОГИИ НЕЙРОННЫХ СЕТЕЙ В ЗАДАЧЕ ОБНАРУЖЕНИЯ ВЗРЫВЧАТЫХ ВЕЩЕСТВ МЕТОДОМ АБСОРБЦИОННОЙ СПЕКТРОСКОПИИ. ВЕСТНИК ВГУ СЕРИЯ: СИСТЕМНЫЙ АНАЛИЗ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ. 2017 г., № 1.
 23. О представлении непрерывных функций нескольких переменных в виде суперпозиции непрерывных функций одного переменного. А.Н., Колмогоров. №5, 1958 г., АН СССР, Т. Т.114, стр. 953 - 956.
 24. СанПиН 2.6.1.3289-15 Гигиенические требования по обеспечению радиационной безопасности при обращении с источниками, генерирующими рентгеновское излучение при ускоряющем напряжении до 150 кВ.
 25. СанПиН 2.6.1.2523-09 Нормы радиационной безопасности (НРБ-99/2009).
 26. СП 2.6.1.2612-10 Основные санитарные правила обеспечения радиационной безопасности (ОСПОРБ 99/2010).
 27. СП 2.2.3670-20 Санитарно-эпидемиологические требования к условиям труда.
 28. ТОИ Р-45-084-01. Типовая инструкция по охране труда при работе на персональном компьютере.
 29. ГОСТ 12.2.032-78 Рабочее место при выполнении работ сидя. Общие эргономические требования.
 30. СП 52.13330.2016 Естественное и искусственное освещение.

31. СанПиН 2.2.4.3359-16 Санитарно-эпидемиологические требования к физическим факторам на рабочих местах.
32. СанПиН 2.2.2776-10 "Гигиенические требования к оценке условий труда при расследовании случаев профессиональных заболеваний".
33. СанПиН 2.2.4-548-96 Гигиенические требования к микроклимату производственных помещений.
34. СН 2.2.4/2.1.8.562-96 Санитарные нормы. Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки.
35. ГОСТ 12.1.003-2014 Система стандартов безопасности труда. Шум. Общие требования безопасности.
36. ГОСТ 12.1.019-2017 Система стандартов безопасности труда ЭЛЕКТРОБЕЗОПАСНОСТЬ.
37. ГОСТ Р 55102-2012 Ресурсосбережение ОБРАЩЕНИЕ С ОТХОДАМИ.
38. N 99-ФЗ Федеральный закон "О лицензировании отдельных видов деятельности".
39. СП 12.13130.2009 ОПРЕДЕЛЕНИЕ КАТЕГОРИЙ ПОМЕЩЕНИЙ, ЗДАНИЙ И НАРУЖНЫХ УСТАНОВОК ПО ВЗРЫВОПОЖАРНОЙ И ПОЖАРНОЙ ОПАСНОСТИ.
40. СП 9.13130.2009 Техника пожарная. Огнетушители.

Список публикаций

1. Петрович Е.В., Семёнов М.Е., Ляпушкин С.В. Метод проверки релевантности рентгенофлуоресцентного спектра применяемой градуировке // АКТУАЛЬНЫЕ ВОПРОСЫ ЭНЕРГЕТИКИ: Материалы Всероссийской научно-практической конференции с международным участием, Омск 14–15 мая 2021 года – Омск: Изд-во ОмГТУ, 2021. – С. 208-211.

Приложение А

Development of a software product for processing X-ray spectra data

Студент:

Группа	ФИО	Подпись	Дата
0ВМ92	Петрович Евгений Витальевич		

Руководитель ВКР:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОЭФ	Семенов М.Е.	к.ф.-м.н., доцент		

Консультант-лингвист отделения иностранных языков ШБИП:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель ОИЯ	Утятина Янина Викторовна	Старший преподаватель		

Introduction

Currently, for the qualitative and quantitative element analysis of the chemical composition of substances in industry, are widely used devices, which principles of operation are based on obtaining and analyzing the characteristic X-ray spectra of these substances. The method is based on Moseley's law, which relates the energy of the spectral lines of characteristic X-ray radiation of an atom of a chemical element with its ordinal number. The advantages of characteristic X-ray chemical analysis over other methods of analysis are as follows:

- relative simplicity;
- relatively low cost;
- non-destructive analysis;
- analysis is possible in any aggregate state of matter;
- low requirements for sample preparation;
- the ability to carry out simultaneous analysis for several elements;
- speed of analysis.

One of the varieties of analytical tools for CXR spectra is the X-ray energy dispersive spectrometers. Their main advantage over mass spectrometers, spectrometers with wave dispersion, and other similar instruments is the relative simplicity of design, reliability, and low cost of the instruments themselves and their maintenance.

The quantitative elemental spectral analysis technique has two main parts:

obtaining and storing the X-ray spectrum of the investigated material using the hardware part of the spectrometer;

processing of the X-ray spectrum on a computer using special software to obtain information on the chemical composition of the sample under study.

There are several different methods for processing X-ray spectra and determining the values of the concentrations of chemical elements in the test material. All of them require primary preprocessing of the spectra, which includes the following steps:

- noise suppression,
- background removal,
- selection of the peaks of lines of the elements presents and determination of their intensities.

The data obtained in this way are used to determine the quantitative elemental composition of the material under study using one of the following methods:

- method of fundamental parameters (MFP);
- methods based on standard samples, usually it is the polynomial regression.

These classical principles and methods of spectral analysis are described in detail in [1] [2] [3] [4] [5].

Thus, the process of quantitative elemental spectrum analysis is a complex step-by-step process. The algorithm of this process is contained in the calibration - a software object that contains:

- the values of the spectrometer settings at which the spectrum should be captured (exposition in seconds, the detector gain factor, etc.),
- the primary preprocessing algorithm, and the mathematical model - the calibration equation that implements the method using which the concentrations are determined.

The calibration creation requires the involvement of experienced highly qualified specialists, whose work consists of the manual selection of all calibration parameters. As a rule, the specialists are involved directly from the manufacturer of the spectrometer or their authorized representatives, which is a factor that significantly complicates and increases the cost of using spectrometers.

At present time, it is being actively carried out researches on the application of the neural networks deep learning technologies for the analysis of spectra for various applied problems [6] [7] [8] [9] [10] [11] [12]. Neural networks have proven themselves well in solving a wide variety of regression and classification problems.

Nevertheless, the analysis of the products of the world's leading manufacturers of spectrometers, such as Shimadzu [13], Bruker [14], PANalytical [15], ThermoFisher Scientific [16], Oxford Instruments Industrial Analysis [17], and others, showed that at the current moment only classical analytical methods are used in their software.

Although deep learning neural networks have been used to solve regression problems for a long time, applying a deep learning neural network to create a calibration for a spectrometer has a feature selection problem [18].

If the intensity values of the selected peaks of the spectrum are supplied to the input of the neural network as features, this does not differ much from the classical methods of spectral analysis. A specialist must isolate features from spectra, select important features, determine the need for normalization, check all features for multicollinearity, etc.

If the values of the intensities in the channels of the spectrum are fed to the input of the neural network, then in this case the features will have strong multicollinearity, since the intensity of each peak of the characteristic radiation is distributed over several channels. In this case, the regression model will have high statistical uncertainty, thus the estimates obtained using such a model will have unacceptable variance.

The purpose of this master's dissertation is to develop, implement as software and study a method for creating calibrations based on the use of deep learning algorithms for neural networks.

Scientific and practical novelty and significance of the work.

The proposed method solves the problem of the intensity multicollinearity in the spectrum channels. Application of the method does not require the involvement of qualified specialists, which entails a significant reduction in the costs of operating the spectrometer, and allows the creation of high-quality calibration by the laboratory personnel. Also, thanks to the ability to create complex calibration, it is possible to expand the areas of use of spectral analysis.

Results of work.

The developed software and the calibrations created using it were presented to specialists of TechnoAnalit LLP, specialists of uranium mining enterprises, and non-ferrous metallurgy enterprises. The work received high scores and currently the software product is being tested in chemical laboratories of mining enterprises in Kazakhstan.

2. The theoretical basis of the research and problem statement

2.1 Calibration of the spectrometer

Each chemical element, exposed to exciting X-ray radiation, emits characteristic radiation with a unique set of spectral lines. The analyzed spectrum is a complex superposition of characteristic radiation lines from all chemical elements contained in the test material, scattered radiation from an X-ray tube and other artifacts. The intensity of the characteristic radiation in the corresponding lines of the spectrum depends on the concentration of chemical elements in the material under study. In addition, the chemical elements in the test material have a mutual effect on the characteristic radiation of each other, weakening or, conversely, amplifying it.

The radiation intensity is expressed as a number of X-ray photons emitted by the material subjected to X-ray irradiation and is a functional relationship:

$$I = \int_{t_0}^{t_0+\tau} N(e, t) dt, \quad (1)$$

where $I \in \mathbb{N}$ - radiation intensity, $N \in \mathbb{N}$ -the rate of count of photons per time unit, $e \in \mathbb{R}$ - energy of photons, $t \in \mathbb{R}$ – time, $\tau \in \mathbb{R}$ - exposition duration.

During the exposition time, the spectrometer records photons of radiation from the studied material and their number is distributed in accordance with their energies on a discretized energy scale. Thus, the measured spectrum can be represented as an interval variation series with an equal interval grouping:

$$S = \{I_n: I_n \in \mathbb{N}, n \in \mathbb{N}, 0 < n < K\} \quad (2)$$

where I_n – the number of the registered photons in the n-th channel, K - the number of intervals in the spectrum (Figure 1).

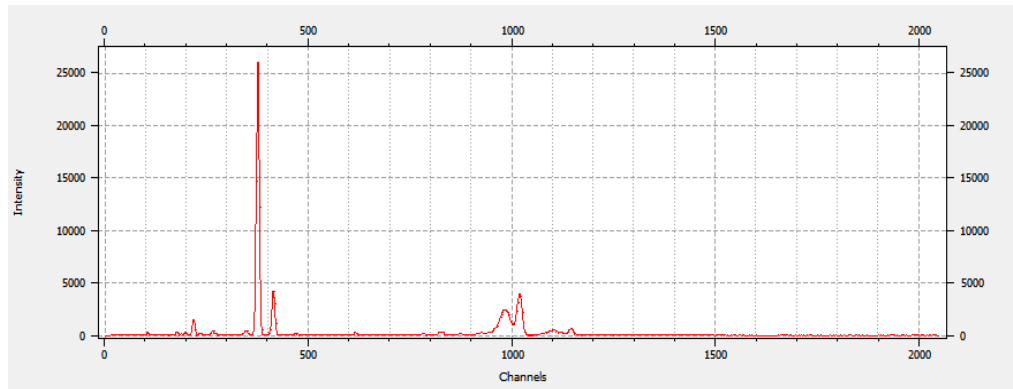


Figure 1 - Representation of the spectrum as a histogram

The concentration of a chemical element is a mapping G of an interval variation series S onto a set of positive real numbers:

$$G: S \rightarrow \mathbb{R}^+. \quad (3)$$

Mapping G - in the spectral analysis is called calibration.

There are several methods for calculating the concentrations of chemical elements. One of the most frequently used, included in the software of almost all modern spectrometers, is the intensity correction method [1] [2].

It was found that the concentration of a chemical element in a light, neutral matrix, that is, such that the chemical elements that make up this matrix do not in any way affect the intensity of X-ray fluorescence of the determined chemical element, is directly proportional to the intensities in the peaks of the X-ray lines of this element:

$$C_A = \frac{n_A}{\eta_A}, \quad (4)$$

where C_A – is the concentration of the determined chemical element, n_A - is the intensity at the peak of the CXR line of the determined chemical element, η_A – is the specific counting rate of the number of pulses in this line when the concentration of the determined element changes by 1%. The η_A value is calculated through the specific intensities at the peak of the X-ray line obtained from samples with known concentrations of the determined chemical element.

Under real conditions, in the overwhelming majority of cases, the count rate η_A is influenced by other chemical elements that make up the chemical composition of the sample under study. In such cases, the calculated value of concentration C_A contains significant deviations from the actual concentration values and, therefore, the measurement accuracy will be unsatisfactory (Figure 2). In addition, the mutual influence of elements on each other is complex and transitive. As shown in Figure 3, this leads to a complex relationship between the intensities at the peaks of the X-ray lines and the concentration of a chemical element.

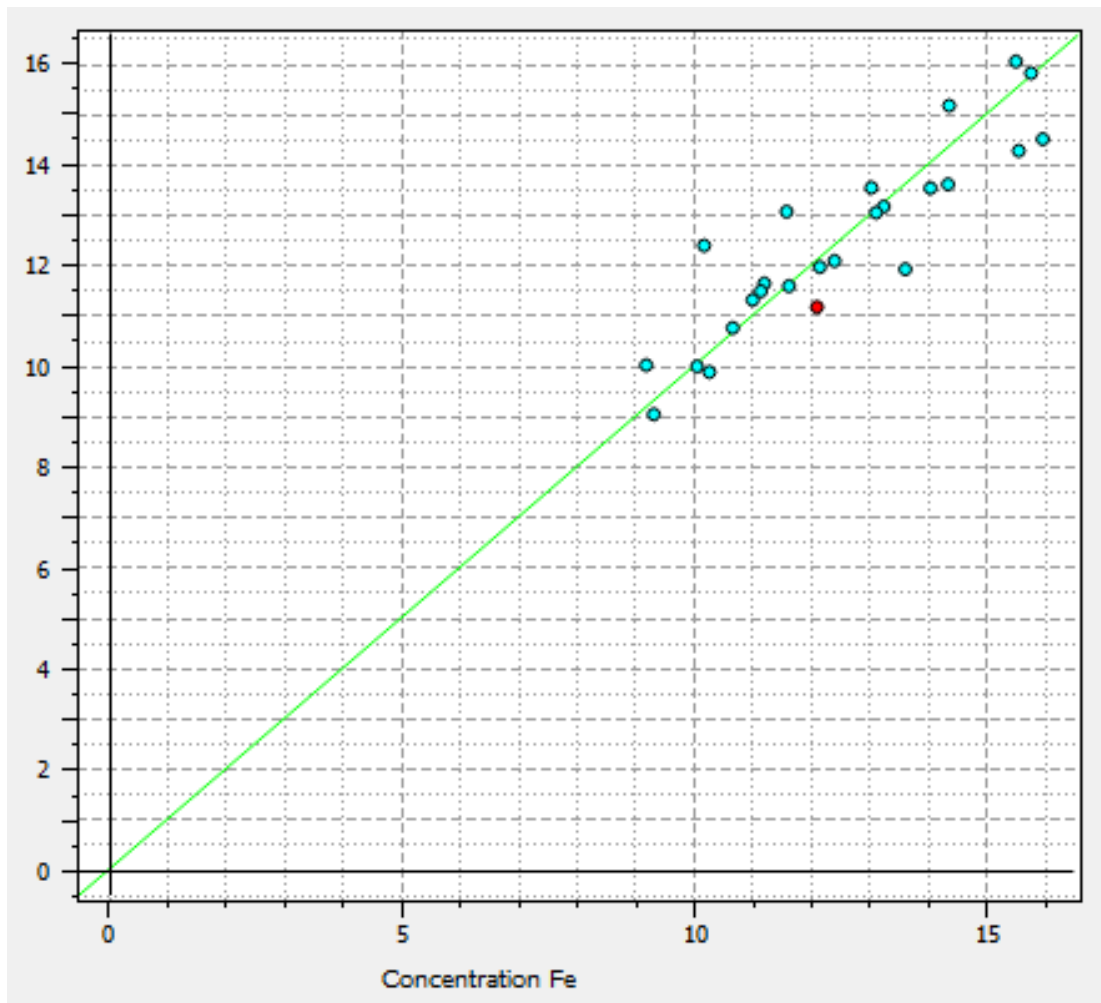


Figure 2 - Calculated Fe concentration without taking into account the influencing chemical elements

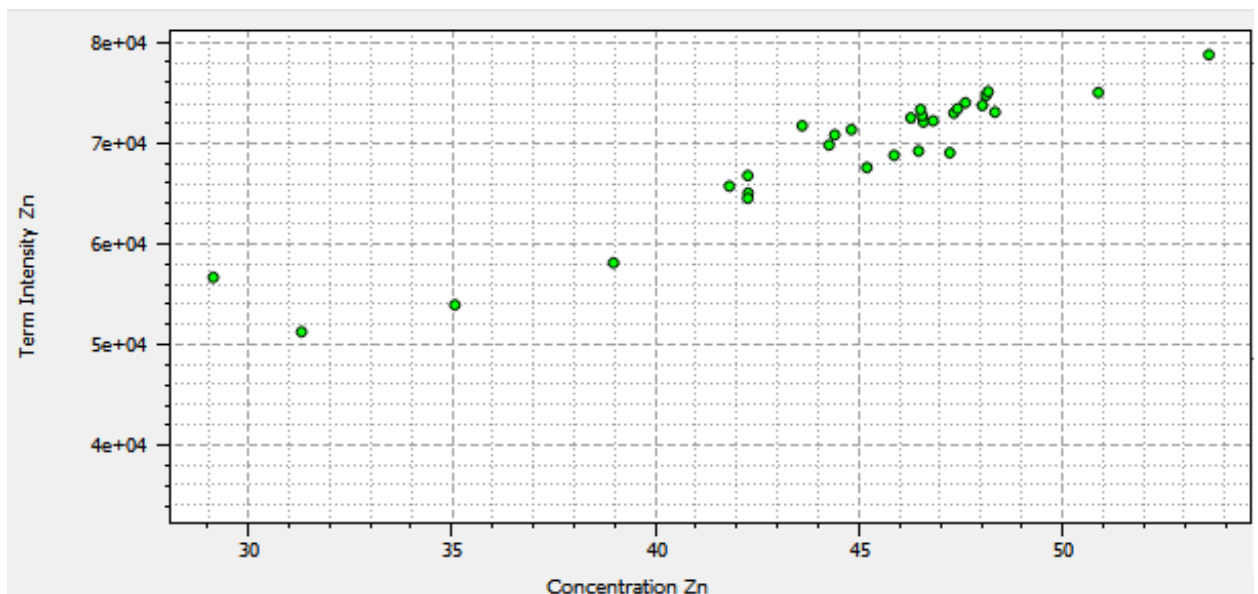


Figure 3 - Complex relationship between the intensity in the X-ray line of Zinc and its concentration in the test material

To correct the effect of other chemical elements, additional correcting variables are introduced into equation (4). In view of the complex chemical composition of real samples and the complex nature of the mutual influence of some chemical elements on others, the concentration equation takes the form of an equation with many independent variables expressing the intensities in the influencing peaks, that is, the concentration of the determined chemical element is a function of the intensities in m peaks of the energy lines:

$$C_A = f(n_1, n_2, \dots, n_m). \quad (5)$$

In most cases, the concentration equation is used in the form of polynomial regression:

$$C_A = a_{A0} + a_{A1}n_A + a_{A2}n_A^2 + \sum_{i=1}^m a_{i1}n_i + \sum_{i=1}^m a_{i2}n_i^2 + \sum_{i=1, j=1, j \neq i}^m a_{i3}n_in_j, \quad (6)$$

where C_A is the concentration of the determined chemical element, n_A is the intensity at the peak of the X-ray line of the determined chemical element, n_i, n_j are the intensity at the peak of the X-ray line of the influencing chemical element, a_i are the coefficients reflecting the degree of influence of the intensity at the peak of the corresponding X-ray line. Taking into account corrective intensities significantly increases the accuracy of concentration calculations (Figure 4).

The independent variables of the regression model are the values of the intensities of the characteristic radiation in the lines of the element, the concentration of which is to be determined, as well as the values of the intensities of the characteristic radiation in the lines of the influencing elements.

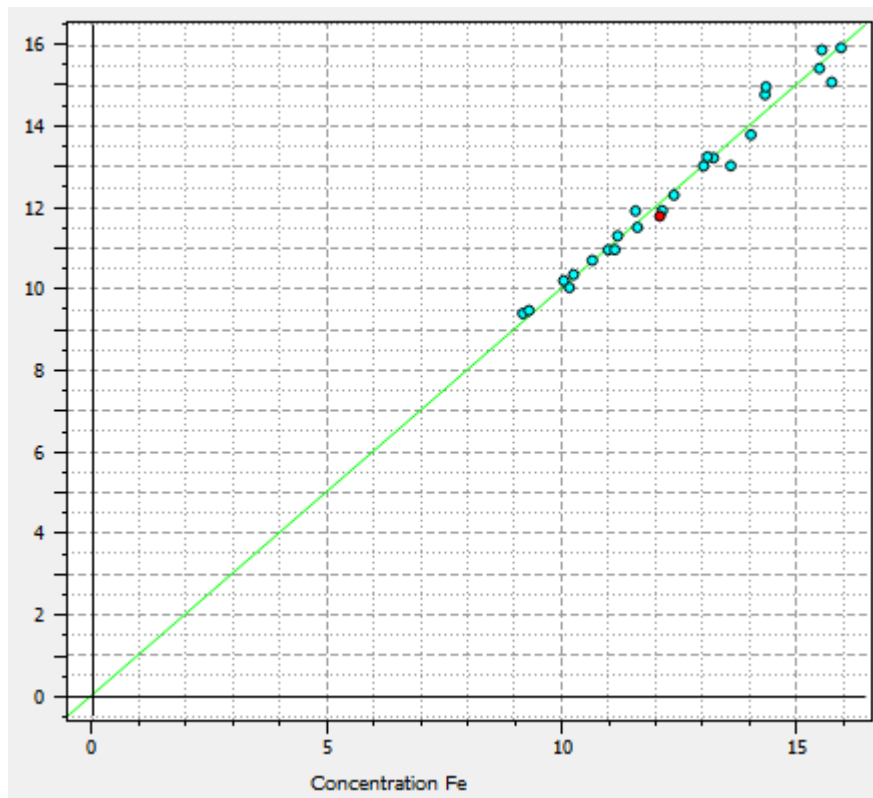


Figure 4 - Calculated Fe concentration value taking into account the influencing chemical elements

The independent variables of the regression model are the values of the intensities of the characteristic radiation in the X-ray lines of the element, the concentration of which is to be determined, as well as the values of the intensities of the characteristic radiation in the lines of the influencing elements.

To create a calibration, the following operations are performed:

- the base peak in the spectra is highlighted - the most pronounced peak of the characteristic radiation line of the element being determined,
- the presence of influencing chemical elements in the test material is determined by the spectrum lines and a decision to include the peaks of influencing elements in the calibration is made,

- the need and type of normalization of the intensity values in the included peaks is determined,
- the need to create artificial predictors in the regression equation, such as the product or the sum of the intensity values in two peaks is determined,
- the regression coefficients are calculated, the quality of the calibration on the calibration and verification sets of spectra is checked.

The correct choice of these parameters critically affects the quality of the calibration and, as a consequence, the accuracy of the analysis results. The process of creating a calibration is iterative, since the mutual influence of elements on each other is complex and transitive.

At each iteration, a new calibration is created with a different set of predictors. Further, from the set of created calibrations, the calibration with the maximum quality indicators is selected - the coefficient of determination R^2 and the standard deviation σ .

The creation of a high-quality calibration is the most difficult operation and is one of the problems associated with the use of spectral analysis.

To solve these problems, it is proposed to apply deep learning algorithms for neural networks to create calibration.

2.2 The theoretical basis of the neural network

Neural networks have several unique properties in comparison with other methods, such as resistance to noise in the input data, the ability to generalize, the ability to approximate the functional dependence of any order of nonlinearity and complexity.

A neural network is a structure consisting of elementary elements - artificial neurons. The structure of the artificial neuron is shown in Figure 5.

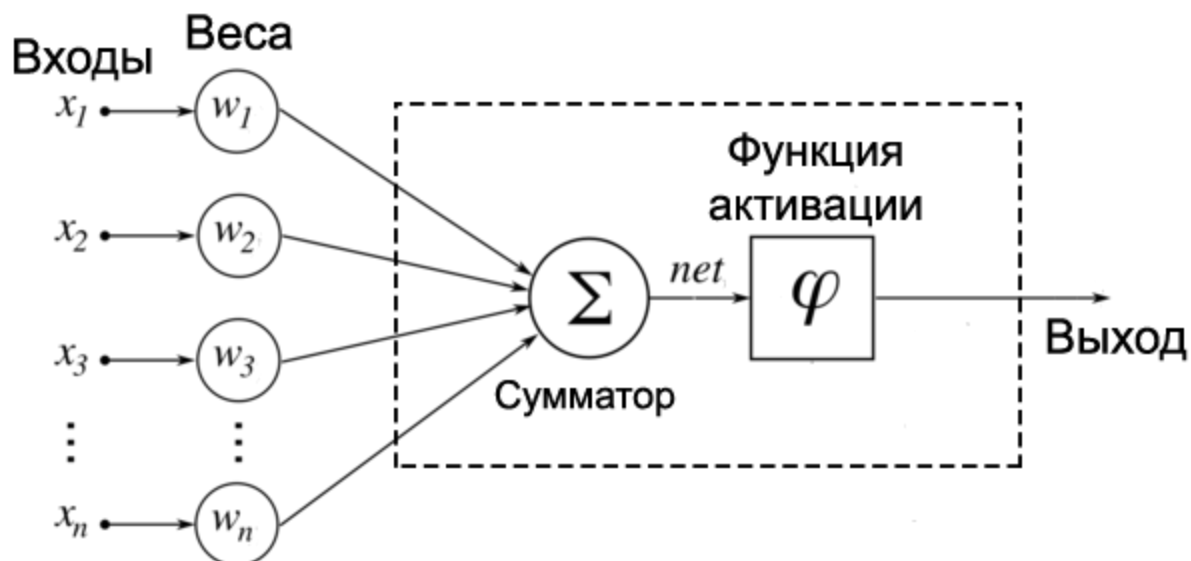


Figure 5 - The structure of an artificial neuron

An n -dimensional vector of some X values is fed to the input of the neuron. Each value x_i is multiplied by the weighting factor w_i and fed to the adder. Next, the sum of the weighted elements of the vector X is fed to the input of the activation function φ .

Thus, the artificial neuron implements the function:

$$y = \varphi(u), u = \sum_{i=1}^n x_i w_i, \quad (7)$$

where φ is the activation function, u is the weighted sum, as the sum of n products of input signals by the corresponding weights.

With a linear activation function, the function (7) is equivalent to a linear regression function. For the implementation of many practical tasks, this is not enough, since the dependencies are, as a rule, non-linear.

By the theorem of Kolmogorov (1957) “Any continuous function of n variables can be represented in the form

$$F(x_1, x_2, \dots, x_n) = \sum_{j=1}^{2n+1} g_j \left(\sum_{i=1}^n \varphi_{ij}(x_i) \right), \quad (8)$$

where g_j and φ_{ij} are continuous functions, and φ_{ij} are independent of the function F [22].

According to theorem (8), by connecting a sufficient number of neurons in series in a network, that is, by a superposition of the functions implemented by them, an arbitrarily complex dependence can be realized with any accuracy.

Usually, to solve regression problems fully connected neural networks with one or more hidden layers are used (Figure 6).

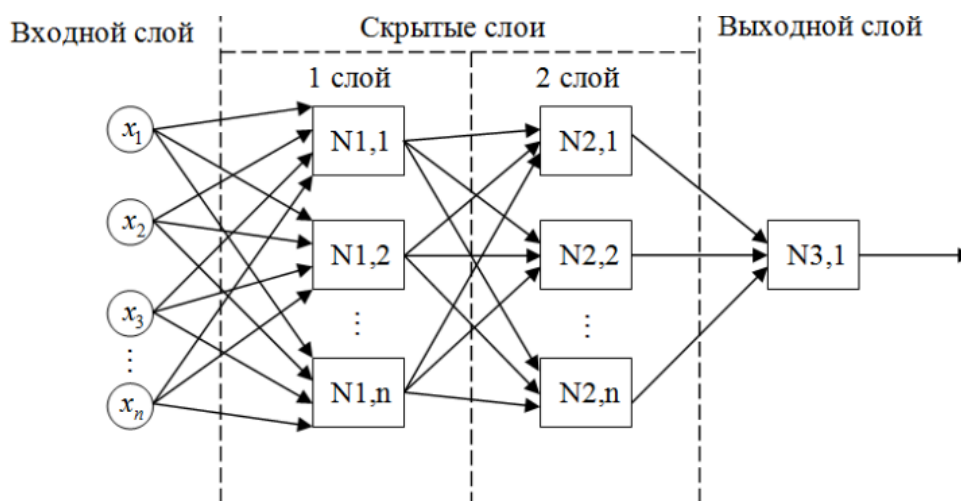


Figure 6 - Block diagram of a multilayer fully connected neural network

The feature vector X which is described the object is passed to the input of the model. The output layer of the neural network is one neuron, at the output of which is the value of the regression function y , implemented by the neural network. The function y implemented by the neural network is determined by the neuron activation functions $\varphi_i(u)$ and the weight coefficients w_{ji} .

The neuron activation function is selected at the neural network design stage, depending on the task. For regression problems, the activation function is usually either linear:

$$f(x) = x, \tag{9}$$

either ReLu:

$$f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \quad (10)$$

The values of the weighting coefficients are adjusted during the training of the neural network.

To train a neural network, many feature vectors of objects are required

$$X = \{x_1, x_2, \dots, x_l\} \quad (11)$$

and the set of vectors of desired neural network responses corresponding to each object

$$Y = \{y_1, y_2, \dots, y_l\} \quad (12)$$

where l is the number of objects in the sample.

Set of pairs

$$X^l = (x_i, y_i)^l \quad (13)$$

called a training sample.

The learning algorithm consists in sequentially feeding the vector of features of the object x_i to the input of the neural network and adjusting the weight coefficients so that the vector of values at the output of the neural network $a(x_i)$ is as close as possible to the vector of the desired responses y_i .

To determine the proximity of vectors, the loss function $L(x_i, w)$, is introduced, where w is the vector of the neural network weight coefficients. For regression problems, the most often used as the loss function is the absolute value of the error

$$L(x_i, w) = |a(x_i, w) - y_i| \quad (14)$$

or squared error

$$L(x_i, w) = (a(x_i, w) - y_i)^2 \quad (15)$$

To assess the quality of a neural network, a quality functional is introduced - empirical risk:

$$Q(X^l, w) = \frac{1}{l} \sum_{i=1}^l L(x_i, w) \quad (16)$$

The learning task is reduced to the empirical risk optimization task:

$$\mu(X^l) = \underset{w}{\operatorname{argmin}} Q(X^l, w). \quad (17)$$

Optimization is performed using the error backpropagation method:

- the weights w are initiated by random small values;
- a pair (x_i, y_i) is randomly selected from the training sample;
- the vector of features x_i is fed to the input of the neural network;
- the local gradient is calculated $\delta_j = L(x_i, w) * \frac{df_j(e)}{de} * w_{ij}$;
- the neural network weights are corrected in the direction from the output to the input according to the formula

$$\Delta w_{ij} = -\eta \delta_j o_i, \quad (18)$$

where o_i is the output of the i -th neuron, $f_j(e)$ is the activation function of the j -th neuron.

Steps 2-5 are repeated until the weights w of the neural network stabilize.

Thus, a neural network with tuned weights and correctly selected activation functions implements the functionality of any kind that maps the input feature vector into the output target value.

2. Development of the software

2.1 Solving the feature selection problem

The use of fully connected neural networks for solving regression problems is widespread. But to create a calibration for a spectrometer in this manner, the use of a neural network has problems. One of the problems is the problem of feature selection.

Figure 7 shows two peaks of the lines of the X-ray fluorescence spectrum.

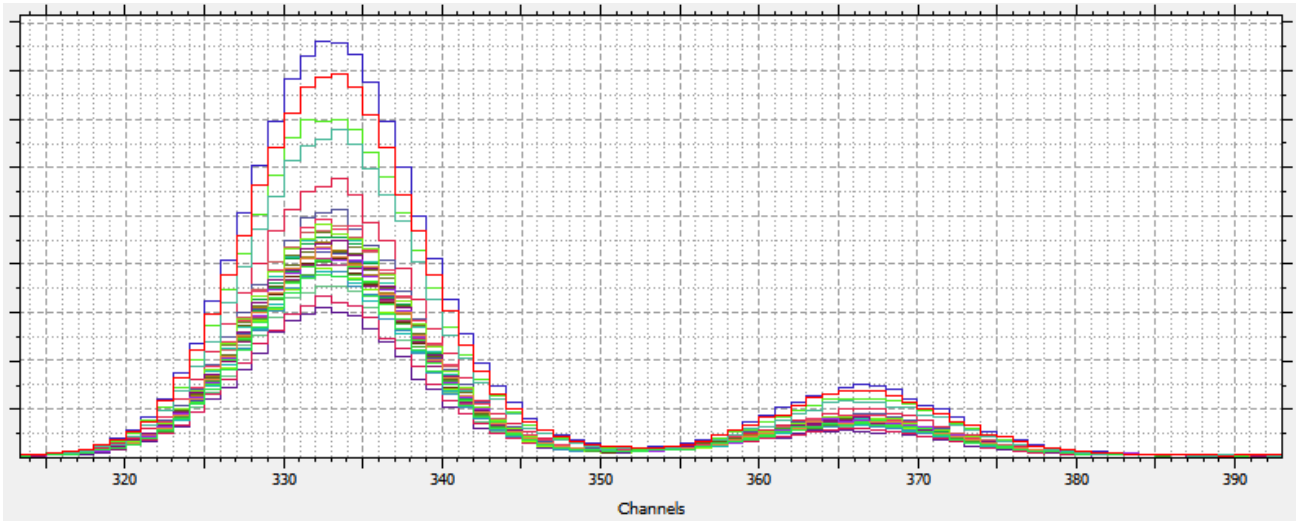


Figure 7 - Multicollinearity of intensity values in channels representing one peak of the X-ray line

Each peak is represented by several tens of channels: the first peak - channels 318 - 350, the second - channels 354 - 380. It is obvious that the intensity values in the channels of each peak have strong multicollinearity. Therefore, the regression model, the predictors of which will be the intensity values in the spectrum channels, will have a high statistical uncertainty. The estimates of such a model will have unacceptable variance values.

As a result of research work to solve this problem, a method for creating calibrations using a convolutional neural network was proposed and implemented.

Convolutional neural networks are widely used in image recognition algorithms.

Convolution is a cross-correlation function of two functions. One of the functions - the original signal $X = \{x_1, x_2, \dots, x_l\}$, the other function w_a , where $a = -A, \dots, +A$, is a filtering one. During the convolution operation, the filtering function w "runs" over the entire input signal and the dot product is calculated at each position

$$(x * w)[i] = \sum_{a=-A}^{+A} w_a x[i + a]. \tag{19}$$

As a result, the positions in the convolution function that have the maximum values indicate the positions in the original signal that are most similar to the filtering function (Figure 8).

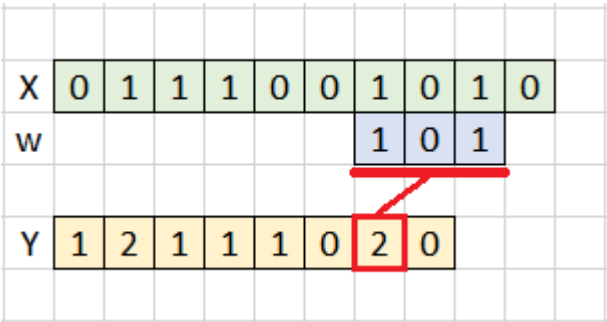


Figure 8 - Scheme of the convolution operation.

In convolutional neural networks, the convolution operation is used to find patterns of input signal features distributed in a vector. The signal in the input layer is scanned by several convolutional filters. Convolution signals are fed to the neurons of the next layer. Each filter forms its own separate group of neurons in the next layer, called a feature map. Significant values in the feature map show the presence of a particular feature at a particular position of the input signal. Each feature map is scanned by the next group of filters that highlight features of even higher abstraction in the maps.

Filters are sets of weights that are adjusted during the training of the convolutional neural network.

In addition to direct convolutions, a pulling or rescaling operation is performed in a convolutional neural network. As with the convolution operation, the feature map

is scanned by a window of a certain size with such a step so that each new scan does not intersect in the previous one. One value is selected from the feature maps, as a rule, the maximum. Thus, a window with a size of two values is reduced by half. This allows you to analyze the image at large scales. In addition, the dimension of the neural network is reduced, which makes it possible to reduce the number of tunable parameters, which speeds up training. A schematic of the pulling operation is shown in Figure 9.

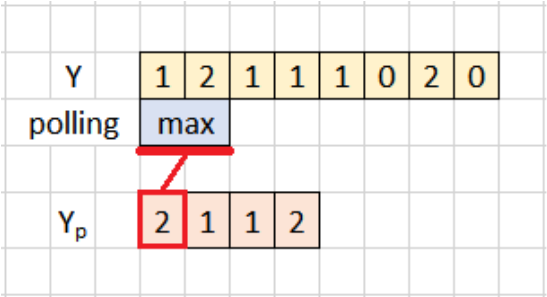


Figure 9 - Scheme of the pulling operation

At the tail of the convolutional neural network, the selected features are aligned and fed into a fully connected neural network in which the final operation is performed, as a rule, the classification of objects present in the image. A schematic of a convolutional neural network for image classification is shown in Figure 10.

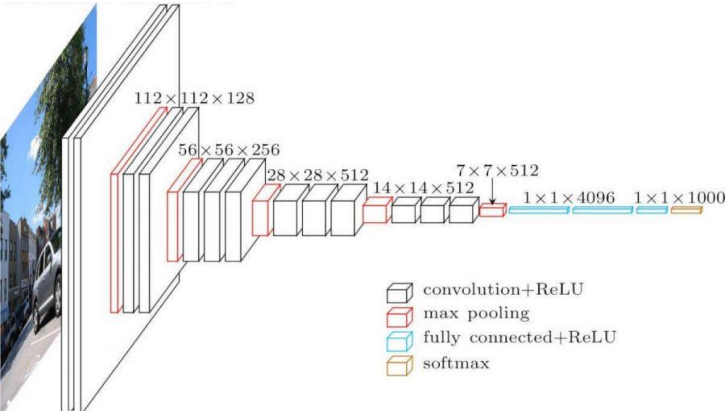


Figure 10 - Scheme of a convolutional neural network VGG16

The X-ray spectrum can be represented as a one-dimensional image. With the help of convolutional layers, features are extracted from the spectrum, which is fed to the input of the fully connected - regression part of the neural network. The regression part has one output neuron with the ReLu activation function suitable for the regression problem. Thus, the problem of multicollinearity of input features is solved.

2.2 Solving the small sample problem

The second problem of using a neural network for spectrometer calibration problems is related to the small size of the training sample.

For high-quality training of neural networks, hundreds and thousands of objects in the training set are required. As a rule, the number of spectra available for creating a calibration does not exceed several tens. Since the spectra of the same material captured on different spectrometers differ significantly from each other due to individual differences in the parameters of the spectrometer hardware components, such as an X-ray tube, a semiconductor X-ray detector, a measuring chamber, etc. of each spectrometer spectra. The use of such a small training sample size leads to an unsatisfactory quality of the calibration model.

In the course of the study, a two-stage neural network training method was proposed to solve this problem. At the first stage, a preliminary training of the neural network is performed on a set of spectra collected from the spectra measured to determine all possible chemical elements and compounds for all possible concentration ranges.

At the second stage, the neural network is additionally trained on a calibration set for the target chemical element and concentration range.

2.3 Selecting the required tools

To carry out the study, the calibration spectra of various materials with a wide range of concentrations of chemical elements were measured on a SRV-1V

spectrometer manufactured by TechnoAnalit LLP. Some sets of calibration spectra are shown in Figure 11.

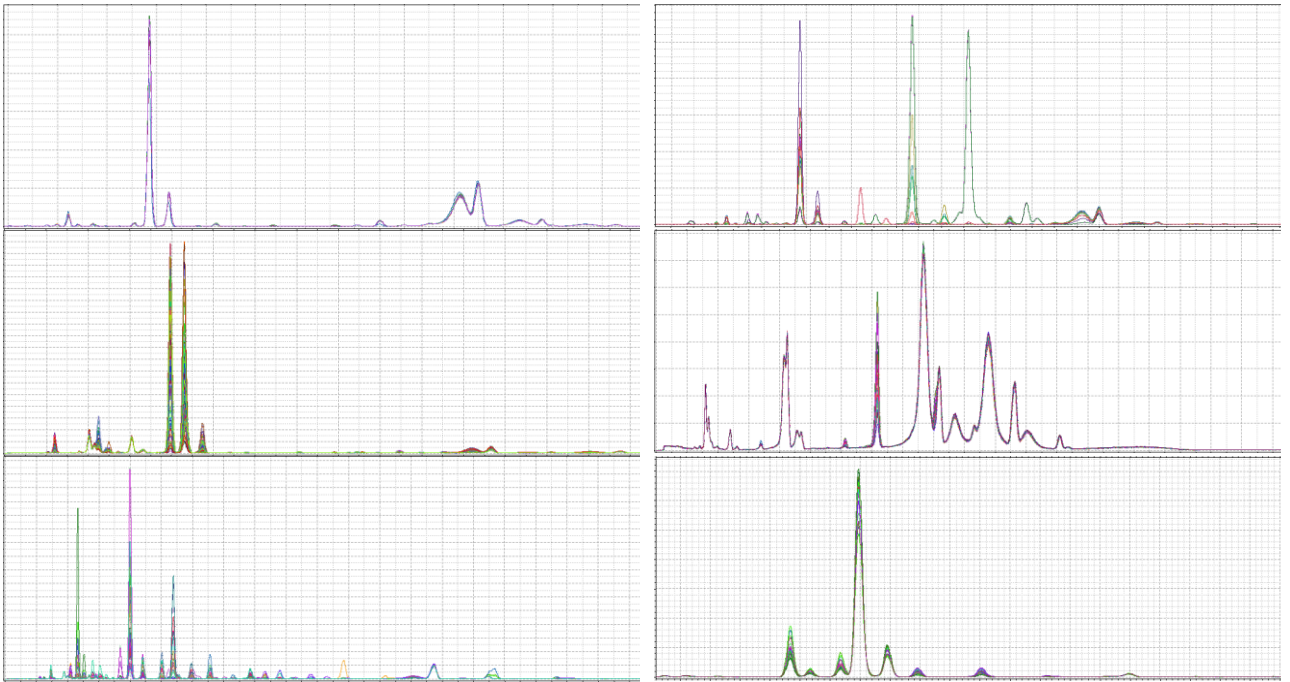


Figure 11 - Sets of calibration spectra

For the implementation of the test version of the software product and computational experiments, the Python 3.6 programming language and the Jupyter Lab programming environment were chosen. The neural network is implemented using the Keras library. The neural network was trained using an RTX 2070 graphics card and CUDA technology, which allows the use of GPU cores to train neural networks.

The following tools were used to integrate the software product into the existing software of TechnoAnalit LLP:

- C ++ programming language,
- Cross-platform framework Qt 5.15.0,
- IDE Qt Creator,
- Integrated into Qt Creator editor of UML models,
- Doxygen is a tool for automatic generation of project documentation,

- Version control system Git.

This toolkit is an internal standard of TechnoAnalit LLP for software development. Software products developed using this suite are designed to meet cross-platform and performance requirements.

Conclusions

In this work on this master's dissertation, a method for creating a spectrometer calibration using deep learning algorithms for a neural network without any preliminary processing of spectra was developed. The features describing the spectra were the values of the radiation intensities in the channels of the spectrum. The difficulties associated with this approach, due to the multicollinearity of features and the small size of training samples, have been overcome. The determination coefficient R^2 , calculated on the test samples of the spectra, was not less than 0.9. As a loss function, the absolute error function was slightly better than the root mean square function. The results obtained allow us to assert that the gradings created using deep neural network algorithms are not inferior in quality to those created by highly qualified specialists. The use of the developed methods allows obtaining a significant economic effect.

The method was implemented as a test module in Python in the Jupyter Lab environment and after testing was integrated into the SRV-1 B spectrometer software. The developed module was tested on real calibration spectra. The graduations obtained using the module are of consistently high quality. Thus, the developed method allows one to obtain calibrations directly from the spectra. Working with the software module does not require highly qualified personnel. The results of the computational experiment were discussed with specialists from TekhnoAnalit LLP, which produces X-ray fluorescence spectrometers, and specialists from non-ferrous metallurgy and uranium mining enterprises. The method was highly appreciated. A program for further

testing of the software module and its further integration into the software of SRV spectrometers manufactured by TechnoAnalit LLP has been developed.

Приложение Б

```
import tensorflow as tf
import numpy as np
import scipy as sp
import pandas as pd
import os
import matplotlib.pyplot as plt
import datetime

#from scipy.misc import toimage
%matplotlib inline

from tensorflow.keras import layers
from tensorflow.keras import Input
from tensorflow.keras import Model
from tensorflow.keras import backend as K
from tensorflow.keras.models import Sequential
from tensorflow.keras.models import model_from_json
from tensorflow.keras.layers import Dense, Flatten, Conv1D, MaxPooling1D,
GlobalMaxPooling1D, Dropout, BatchNormalization, Activation, LeakyReLU
from tensorflow.keras.regularizers import Regularizer
from tensorflow.keras import regularizers
from tensorflow.keras.initializers import GlorotNormal, GlorotUniform, Orthogonal

from tensorflow.keras.losses import mse, mae, msle, mape, binary_crossentropy
from tensorflow.keras.metrics import Accuracy
from tensorflow.keras.optimizers import Adam, SGD, RMSprop, Nadam, Adadelta,
Adagrad
from tensorflow.keras.regularizers import l2

from sklearn.model_selection import KFold
from sklearn.metrics import r2_score

def l1_reg_(weight_matrix):
    return 0.00008 * tf.math.reduce_sum(tf.math.abs(weight_matrix))
def l1_reg(weight_matrix): # L2 regularization
    return 0.00008 * tf.math.reduce_sum(tf.square(weight_matrix))

A_num_folds = 2
B_num_folds = 5

A_l_rate = 0.0001
A_batch_size = 16
```



```

A_loss = binary_crossentropy
A_optimizer = Adam(A_l_rate)
#optimizer = Adadelta(l_rate)
#A_metrics=binary_crossentropy
A_metrics=Accuracy
A_num_epochs = 200
A_verbosity = 1

B_l_rate = 0.00001
B_batch_size = 5
B_loss = mae #msle #mse #mae
B_optimizer = Adam(B_l_rate)
#optimizer = Adadelta(l_rate)
B_metrics=mse
B_num_epochs = 1500
B_verbosity = 1

#=====
# creation of calibration spectra data array and known concentration array
def loadSpeData(dirPath, chem):

    speData = np.array([])
    concData = np.array([])

    for file in os.listdir(dirPath):
        if file.endswith(".spe"):
            print(file)
            # loading spe intensity data
            f = open(os.path.join(dirPath, file))
            lines = f.readlines()
            speData = np.append(speData, list(map(float, lines[20:])))

        if file.endswith(".csv"):
            # loading concentration data
            f = open(os.path.join(dirPath, file))
            concData = np.array(pd.read_csv(f, sep=',')[chem])

    # reshape the spe data according to the number of the spectrum channels
    #speData = np.log(speData + 1)

    mX = np.max(speData)
    speData = speData / mX
    speData = speData.reshape(-1, len(lines[20:]))
    # speData = speData.reshape(-1, len(lines[20:]))
    # reshape the conc data into a single column array
    mY = np.max(concData)
    concData = concData / mY
    concData = concData.reshape(-1, 1)

```

```

# prepare resulting arrays
res = list()
res.append(speData)
res.append(concData)

return res
#=====
def loadModelSpeData(dirPath):
    speData = np.array([])
    concData = np.array([])

    for file in os.listdir(dirPath):
        if file.endswith(".spe"):
            # loading spe intencity data
            f = open(os.path.join(dirPath, file))
            lines = f.readlines()
            #speData = np.append(speData, list(map(float, lines[205:280])))
            speData = np.append(speData, list(map(float, lines[20:])))
            concData = np.append(concData, [sum(list(map(float, lines[205:225])))])

    # reshape the spe data according to the number of the spectrum channels
    speData = speData.reshape(-1, len(lines[20:]))
    mX = np.max(speData)
    speData = speData / mX
    # reshape the conc data into a single column array
    mY = np.max(concData)
    concData = concData / mY
    concData = concData.reshape(-1, 1)

    # prepare resulting arrays
    res = list()
    res.append(speData)
    res.append(concData)

    return res

#=====
def loadEncoderTrainSpeData(dirPath):
    speData = np.array([])
    concData = np.array([])

    for file in os.listdir(dirPath):
        if file.endswith(".spe"):
            # loading spe intencity data
            f = open(os.path.join(dirPath, file))
            lines = f.readlines()
            speData = np.append(speData, list(map(float, lines[20:])))

```

```

# reshape the spe data according to the number of the spectrum channels
speData = speData.reshape(-1, len(lines[20:]), 1)
mX = np.max(speData)
speData = speData / mX

return speData
#=====
def createAutoEncoderModel(input_shape):

    initializer = Orthogonal()
    inputTensor = Input(shape=(input_shape[1:]))

    x = layers.Conv1D(256, (3), activation='relu', padding='same',
kernel_initializer=initializer)(inputTensor)
    x = layers.Conv1D(256, (3), activation='relu', padding='same',
kernel_initializer=initializer)(inputTensor)
    x = layers.MaxPooling1D((2), padding='same', strides = 2)(x)

    x = layers.Conv1D(128, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.Conv1D(128, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.MaxPooling1D((2), padding='same', strides = 2)(x)

    x = layers.Conv1D(64, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.Conv1D(64, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.MaxPooling1D((2), padding='same', strides = 2)(x)

    x = layers.Conv1D(32, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.Conv1D(32, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.MaxPooling1D((2), padding='same', strides = 2)(x)

    x = layers.Conv1D(16, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.Conv1D(16, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)

    encoded = layers.MaxPooling1D((2), padding='same', strides = 2)(x)

    x = layers.Conv1D(16, (3), activation='relu', padding='same',
kernel_initializer=initializer)(encoded)

    x = layers.Conv1D(16, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)

```

```

    x = layers.Conv1D(16, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.Conv1D(16, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.UpSampling1D((2))(x)

    x = layers.Conv1D(32, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.Conv1D(32, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.UpSampling1D((2))(x)

    x = layers.Conv1D(64, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.Conv1D(64, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.UpSampling1D((2))(x)

    x = layers.Conv1D(128, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.Conv1D(128, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.UpSampling1D((2))(x)

    x = layers.Conv1D(256, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.Conv1D(256, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)
    x = layers.UpSampling1D((2))(x)

    decoded = layers.Conv1D(1, (3), activation='relu', padding='same',
kernel_initializer=initializer)(x)

    autoencoder = Model(inputTensor, decoded)
    autoencoder.compile(optimizer=A_optimizer, loss=A_loss)

    #print("INPUT SHAPE", list(autoencoder.input_shape))
    return autoencoder
#=====
def createRegressionModel(base_model):

    model = Sequential()
    model.add(Input(shape = list(base_model.input_shape)[1:], name = "B_Input"))
    for layer in base_model.layers[0:15]:
        layer.trainable = False
        model.add(layer)

    initializer = Orthogonal()

```

```

#initializer = GlorotNormal()

    model.add(Dense(2048, activation='relu', name = "Dense_B_118",
kernel_initializer=initializer, kernel_regularizer=l1_reg))
    model.add(Dense(2048, activation='relu', name = "Dense_B_120",
kernel_initializer=initializer))

    model.add(Dense(1, name = "Dense_B_200", activation='relu',
kernel_initializer=initializer))

    model.compile(optimizer=B_optimizer, loss=B_loss, metrics=B_metrics)

    return model
#=====
def standardize(data):
    # input data standardization
    stdInput = data[0]
    meanList = stdInput.mean(axis = 0)
    stdList = stdInput.std(axis = 0)
    stdInput -= meanList
    stdInput /= stdList
    # target data standardization
    stdTarget = data[1]
    meanVal = stdTarget.mean(axis = 0)
    stdVal = stdTarget.std(axis = 0)
    stdTarget -= meanVal
    stdTarget /= stdVal

    res = list([])
    res.append(stdInput)
    res.append(meanList)
    res.append(stdList)

    res.append(stdTarget)
    res.append(meanVal)
    res.append(stdVal)

    return res

#=====
def generateData():
    import random as rd
    #x = list([])
    # x.append(7)
    # x
    data = np.array([])

```

```

target = np.array([])
for n in range(50):
    x = list([])
    for i in range(5):
        x = x + [rd.random()]
    y = (x[0]**2 + x[1]**3 + x[2] + x[3] + x[4])/5

    data = np.append(data, x)
    target = np.append(target, y)
    # speData = np.append(speData, list(map(int, lines[20:])))

data = data.reshape(-1,5)

res = list()
res.append(data)
res.append(target)

return res
#=====
def loadEncoderModelFromFile(fileName):

    # load json and create model
    json_file = open(fileName + '.json', 'r')
    loaded_model_json = json_file.read()
    json_file.close()
    loaded_model = model_from_json(loaded_model_json)
    # load weights into new model
    loaded_model.load_weights(fileName + ".h5")
    print("Loaded model from disk")

    # evaluate loaded model on test data
    # loaded_model.compile(loss='binary_crossentropy', optimizer='rmsprop',
metrics=['accuracy'])
    loaded_model.compile(optimizer=A_optimizer, loss=A_loss)

    return loaded_model

physical_devices = tf.config.list_physical_devices('GPU')
tf.config.experimental.set_memory_growth(physical_devices[0], True)

# # Define the K-fold Cross Validator
# kfold = KFold(n_splits=num_folds, shuffle=True)
# # K-fold Cross Validation model evaluation
# fold_no = 1
A_dirPath = "Common"
B_dirPath = "Au-1"
B_chemical = "Au"

```

```

#B_dirPath = "Cu-1"
#B_chemical = "Cu"

models = ["A_model-1", "B"]

# for train, test in kfold.split(inputs): #(inputs, targets):
## # Define the model architecture
if "A" in models:

    A_trainData = loadEncoderTrainSpeData(A_dirPath)
    # Define the K-fold Cross Validator
    kfold = KFold(n_splits=A_num_folds, shuffle=True)
    # K-fold Cross Validation model evaluation
    fold_no = 1
    for train, test in kfold.split(A_trainData): #(inputs, targets):
        # # Define the model architecture
        A_model = createAutoEncoderModel(A_trainData.shape)
        ## JUMP

        # Generate a print
        print('-----')
        print(f'Training for fold {fold_no} ...')
        print(A_model.summary())

        # Fit data to model
        A_history = A_model.fit(A_trainData,
                               A_trainData,
                               batch_size=A_batch_size,
                               epochs=A_num_epochs,
                               shuffle=True,
                               #validation_data=(A_trainData[test], A_trainData[test]),
                               validation_split=0.20,
                               verbose=A_verbosity)

        A_model_json = A_model.to_json()
        jsonFileName = "A_model-"+str(fold_no)+".json"
        with open(jsonFileName, "w") as json_file:
            json_file.write(A_model_json)
        # serialize weights to HDF5
        h5FileName = "A_model-"+str(fold_no)+".h5"
        A_model.save_weights(h5FileName)
        print("Saved model to disk")

    A_pred = A_model.predict(A_trainData[test])
    A_trgs = A_trainData[test]

```

```

#print(pred)
#print(targets[test])
A_errList = A_pred - A_trgs
A_err = np.mean(abs(A_pred - A_trgs))
#print(errList)

plt.plot(A_history.history['loss'],
         label='Train loss')
plt.plot(A_history.history['val_loss'],
         label='Test loss')
plt.xlabel('Epoch')
plt.ylabel('Loss')
plt.legend()
plt.show()

# Разброс предсказаний может показать перекося, если есть
plt.scatter(A_trgs, A_pred)
plt.xlabel('Targets')
plt.ylabel('Predictions')
plt.axis('equal')
plt.xlim(plt.xlim())
plt.ylim(plt.ylim())
plt.plot([-100, 100], [-100, 100])
plt.show()

## plot
n = 4
plt.figure(figsize=(50, 10))
for i in range(1, n + 1):
    # Display original
    ax = plt.subplot(2, n, i)
    plt.plot(A_trgs[i])
    #plt.gray()
    ax.get_xaxis().set_visible(False)
    ax.get_yaxis().set_visible(False)

    # Display reconstruction
    ax = plt.subplot(2, n, i + n)
    plt.plot(A_pred[i])
    #plt.gray()
    ax.get_xaxis().set_visible(False)
    ax.get_yaxis().set_visible(False)
plt.show()
fold_no = fold_no + 1

## END TRAIN A MODEL
if "B" in models:
    A_trainData = loadSpeData(B_dirPath, B_chemical)

```



```

#trainData = loadModelSpeData(dirPath)
#trainData = generateData()
# data standardization
#standardizationRes = standardize(trainData)

inputs = A_trainData[0]
inputs = np.expand_dims(inputs, axis = 2)
targets = A_trainData[1]

# Define the K-fold Cross Validator
kfold = KFold(n_splits=B_num_folds, shuffle=True)
# K-fold Cross Validation model evaluation
fold_no = 1
for train, test in kfold.split(inputs): #(inputs, targets):

## get A_model from file
if "A" not in models:
    print("Train " + models[0] + " - B")
    ## load A_model from file
    A_model = loadEncoderModelFromFile(models[0])
    #A_model.trainable = False
    print(A_model.summary())

else:
    print("Train A-B")

## A_model evaluation
A_score = A_model.evaluate(inputs[test], inputs[test], verbose=0)
print("%s: %.2f%%" % (A_model.metrics_names[0], A_score * 100))

## POINT 1

B_model = createRegressionModel(A_model)
print("MODEL B\n", B_model.summary())

## Model Visualisation
import visualkeras
visualkeras.layered_view(B_model).show() # display using your system viewer
visualkeras.layered_view(B_model, to_file='output.png') # write to disk
visualkeras.layered_view(B_model, to_file='output.png').show() # write and show
visualkeras.layered_view(B_model)
## End visualisation

B_history = B_model.fit(inputs[train],
                        targets[train],
                        batch_size=B_batch_size,
                        epochs=B_num_epochs,
                        shuffle=True,

```

```

        validation_data=(inputs[test], targets[test]),
        verbose=B_verbosity)

B_TestPred = B_model.predict(inputs[test])
B_pred = B_model.predict(inputs)

#print(B_pred)
B_TestTrgs = targets[test]
B_trgs = targets
#print(pred)
#print(targets[test])
B_errList = B_TestPred - B_TestTrgs

B_err = np.mean(abs(B_TestPred - B_TestTrgs))
#print(errList)

## B_model evaluation
print("*****")
print("MODEL EVALUATION")
B_score = B_model.evaluate(inputs[test], targets[test], verbose=0)
print("%s: %.2f%%" % ( str(B_model.metrics_names[1]), B_score[1]*100))
R_square = r2_score(B_TestTrgs, B_TestPred)
print('Coefficient of Determination', R_square)

plt.figure(figsize=(12, 8))
plt.plot(B_history.history['loss'],
        label='Train loss')
plt.plot(B_history.history['val_loss'],
        label='Test loss')
plt.xlabel('Epoch')
plt.ylabel('Loss')
plt.legend()
plt.show()

## SCATTER PLOT
# Test
plt.figure(figsize=(12, 8))
plt.scatter(B_TestTrgs, B_TestPred)
plt.xlabel('Targets')
plt.ylabel('Predictions')
plt.axis('equal')
plt.xlim(plt.xlim())
plt.ylim(plt.ylim())
plt.plot([-100, 100], [-100, 100])
plt.show()

# whole srray

```

```
plt.figure(figsize=(12, 8))
plt.scatter(B_trgs, B_pred)
plt.xlabel('All Targets')
plt.ylabel('All Predictions')
plt.axis('equal')
plt.xlim(plt.xlim())
plt.ylim(plt.ylim())
plt.plot([-100, 100], [-100, 100])
plt.show()
```

```
## HISTOGRAM
## Test
# Absolute error
B_error = B_TestPred - B_TestTrgs
plt.figure(figsize=(12, 8))
plt.hist(abs(B_error), bins = 25)
plt.xlabel("Absolute error")
plt.ylabel("Amount")
plt.show()
```

```
relerror = (B_TestPred - B_TestTrgs)/B_TestTrgs
```

```
#Relative error
plt.figure(figsize=(12, 8))
plt.hist(abs(relerror), bins = 25)
plt.xlabel("Relative error")
plt.ylabel("Amount")
plt.show()
```

```
## whole array
B_error = B_pred - B_trgs
plt.figure(figsize=(12, 8))
plt.hist(abs(B_error), bins = 25)
plt.xlabel("All Absolute error")
plt.ylabel("All Amount")
plt.show()
```

```
relerror = (B_pred - B_trgs)/B_trgs
```

```
#Relative error
plt.figure(figsize=(12, 8))
plt.hist(abs(relerror), bins = 25)
plt.xlabel("All Relative error")
plt.ylabel("All Amount")
plt.show()
```

```
fold_no = fold_no + 1
```

Приложение В

Расчет защитного заземления.

Для расчета заземления определяются исходные данные:

- напряжение питания – 220В;
- мощность сетевого источника питания – не более 100 кВА;
- электроды вертикальные, уголкового, с шириной полки – 40мм;
- длина электрода – 2 м;
- глубина размещения электродов – 0,7 м;
- отношение расстояния между электродами к их длине – 2;
- размеры контура заземления L_1 – 24 м, L_2 – 8 м;
- ширина горизонтального полосового электрода – 12 мм;
- тип грунта – глина (удельное сопротивление $\rho_{гр}$ – 40 Ом * м);
- климатическая зона – 2 (средняя низшая температура – -14, средняя высшая температура – -18, повышающий коэффициент для вертикальных заземлителей $K_{пвз}$ – 1,8, повышающий коэффициент для горизонтальных заземлителей $K_{пгз}$ – 4,5).

Расчетный ток замыкания на землю.

Для установок напряжением до 35 кВ., не имеющих компенсирующих аппаратов, принимается $I_{кз} = 1,5 * I_{рз}$, где $I_{рз}$ - ток релейной защиты.

При $I_{рз} = 25$ А, $I_{кз} = 1,5 * 25 = 37,5$ А.

Требуемое сопротивление заземляющего устройства для установок до 1000 В и мощности сетевого источника питания не более 100 кВА принимается $R_d = 10$ Ом.

Суммарная длина горизонтального электрода $l_1=2*(L_1 + L_2) = 2*(24+8) = 64\text{м}$

Расстояния между электродами должно быть не менее 3 м. потому количество вертикальных электродов $n = 20$ шт.

Расчетное значение удельного сопротивления грунта для вертикального заземлителя: $\rho_{вз} = \rho_{гр} \cdot K_{пвз} = 40 \cdot 1,8 = 72 \text{ Ом} \cdot \text{м}$,

для горизонтального заземлителя: $\rho_{гз} = \rho_{гр} \cdot K_{пгз} = 40 \cdot 4,5 = 180 \text{ Ом} \cdot \text{м}$.

Сопротивление одиночного вертикального заземлителя $R_в$ определяется по формуле:

$$R_в = 0,366 \frac{\rho}{l} \left(\lg \frac{2i}{d} + 0,51 \lg \frac{4h_1+l}{4h_1-l} \right) \text{ Ом},$$

$$R_в = 0,366 \frac{72}{2} \left(\lg \frac{2 \cdot 2}{0,95 \cdot 0,04} + 0,51 \lg \frac{4(1+0,7)+2}{4(1+0,7)-2} \right) = 28,4 \text{ Ом}.$$

Сопротивление вертикального заземлителя $R_г$ определяется по формуле:

$$R_г = 0,366 \frac{\rho}{l} \lg \frac{l^2}{d \cdot h} \text{ Ом}, \quad R_г = 0,366 \frac{180}{64} \lg \frac{64^2}{0,5 \cdot 0,012 \cdot 0,7} = 6,16 \text{ Ом}.$$

Расчетное сопротивление заземлителя $R_з$ вычисляется по формуле:

$R_з = \frac{R_в R_г}{R_г \eta_г + R_в \eta_в n}$ Ом, где $\eta_в$ и $\eta_г$ – табличные значения коэффициентов использования заземлителей.

$$R_з = \frac{28,4 \cdot 6,16}{6,16 \cdot 0,32 + 28,4 \cdot 0,64 \cdot 20} = 0,47 \text{ Ом}.$$

Так как условие $R_з \leq R_д$ выполняется, параметры рассчитанного заземления удовлетворяют нормам ГОСТ 12.1.019-2017 [36].