

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«Национальный исследовательский ядерный университет «МИФИ»
(НИЯУ МИФИ)

**ОТЧЕТ О ПРОХОЖДЕНИИ
УЧЕБНОЙ ПРАКТИКИ (ПРОЕКТНО-ТЕХНОЛОГИЧЕСКОЙ)
1 семестр**

Тема: Разработка пользовательского интерфейса для системы обработки радиосигналов с помощью нейросетевых алгоритмов

Практикант	Корнилов Артем Николаевич
Группа	М21-502
Направление подготовки	09.04.01 ИВТ
Место выполнения практики	НИЯУ МИФИ
Руководитель практики от НИЯУ МИФИ	Заева Маргарита Анатольевна, к.т.н., доцент НИЯУ МИФИ
Руководитель практики от предприятия	—

Москва, 2021 г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«Национальный исследовательский ядерный университет «МИФИ»
(НИЯУ МИФИ)

ИНСТИТУТ ИНТЕЛЛЕКТУАЛЬНЫХ КИБЕРНЕТИЧЕСКИХ СИСТЕМ
КАФЕДРА «КОМПЬЮТЕРНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ» (№12)
**09.04.01 Информатика и вычислительная техника, программа «Защищенные
автоматизированные системы обработки информации и управления»**

ГРУППА М21-502

УТВЕРЖДАЮ

Руководитель программы

Ю. Г. Древс

ЗАДАНИЕ НА УЧЕБНУЮ ПРАКТИКУ (ПРОЕКТНО-ТЕХНОЛОГИЧЕСКУЮ)
семестр 1

Тема: Разработка пользовательского интерфейса для системы обработки
радиосигналов с помощью нейросетевых алгоритмов

Практикант:	Корнилов Артем Николаевич, Akornilov1999@mail.ru
	ФИО, контакты
Руководитель практики от НИЯУ МИФИ:	Заева Маргарита Анатольевна, к.т.н., доцент НИЯУ МИФИ, MAZayeva@mephi.ru
	ФИО, звание, степень, должность, контакты
Руководитель практики от предприятия:	
	ФИО, звание, степень, организация, должность, контакты
Дата выдачи задания:	01.11.2021
Дата защиты практики:	в соответствии с графиком защиты

1. Цели и задачи практики:

- 1.1. Изучение предметной области, связанной с темой практики: анализ рекомендованных источников (п.5).
 - 1.2. Получение практических навыков проектирования и реализации решений по теме практики: реализация пользовательского интерфейса для ОС Windows.
 - 1.3. Освоение специальных инструментальных средств для выполнения практики (пп.2.2): Освоение языка Python версии 3. 7. 12 и выбор среды разработки.
 - 1.4. Подготовка заключения по результатам разработки.
-

2. Исходные данные (ограничения):

- 2.1 Язык программирования Python версии 3. 7. 12
 - 2.2 Средства вычислительной техники/специальное оборудование: ПК с операционной системой Windows, поддерживаемой выбранную среду разработки
 - 2.3. Набор параметров системы обработки радиосигналов
-

План работ (1 семестр):

№ п/п	Содержание работ	Срок выполнения	Отчетный материал	Вид контроля
1	Выбор темы НИР и постановка задачи	4 неделя	Раздел отчета о прохождении практики	Текущий контроль: оценки руководителя и аттестация раздела-АР
2	Обзор источников, связанных с темой практики	7 неделя	Раздел отчета о прохождении практики	Текущий контроль: оценки руководителя (консультанта)
3	Работа над теоретической частью: определение требований и основных проектных решений по теме практики	11 неделя	Раздел отчета о прохождении практики, Рабочие материалы: Выбор среды разработки и проектирование интерфейса. Представление интерфейса пользователя приложения и используемых средств ПО	Текущий контроль: оценки руководителя (консультанта)
4	Практическая часть: реализация основных проектных решений, тестирование, отладка, анализ результатов, полученных на практике	14 неделя	Разделы отчета о прохождении практики. Рабочие материалы: исходный код приложения, дизайн приложения.	Текущий контроль: оценки руководителя (консультанта)
5	Подготовка к защите практики	15-16 неделя	Отчет о прохождении практики (Отчет о НИР), Дневник практиканта, Иллюстрации к докладу	АР по письменному отзыву руководителя (консультанта)
6	Защита практики	в соответствии с графиком защиты	Отчет о прохождении практики (Отчет о НИР), Дневник практиканта, Иллюстрации к докладу	Отзывы руководителя, (консультанта). Оценка комиссии за защиту. Итоговая оценка за УП (НИР)

3. Отчетные материалы:

3.1. Дневник практиканта

3.2. Отчет о прохождении практики (оформляется в форме Отчета о НИР, оформление по ГОСТ 7.32-2017, оригинал задания подшивается после титульного листа, объем – 30-40 стр.)

3.3. Иллюстрации к докладу (титул - 1 слайд; исходные данные (ограничения) - 1-2 слайда; цели и задачи – 1 слайд; обзор прикладной области - 1-2 слайда; теоретические сведения - 2-4 слайда; практические результаты - 2-4 слайда; заключение - 1-2 слайда)

4. Рекомендуемые источники:

1. Лутц М. «Изучаем Python», 4-ое издание - Пер. с англ. - СПб: СимволПлюс, 2011 - 1280с.

2. StackOverflow [Электронный ресурс]: <https://ru.stackoverflow.com>3. Документация по Qt: [Электронный ресурс]. URL: <https://doc.qt.io>4. Документация по Matplotlib: [Электронный ресурс]. URL: <https://matplotlib.org/stable/tutorials/index>

Практикант

Корнилов Артем
Николаевич

Подпись

ФИО

Руководитель от НИЯУ МИФИ

Заева Маргарита
Анатольевна

Подпись

ФИО

Руководитель от предприятия

Подпись

ФИО

РЕФЕРАТ

Отчет по прохождении учебной практики содержит 29 страниц текстового документа формата А4, включающего 14 рисунков, 1 таблицу, 8 использованных источников.

PYTHON, PYQT, MATPLOTLIB, IDE PYCHARM, СРЕДА РАЗРАБОТКИ, ПОЛЬЗОВАТЕЛЬСКИЙ ИНТЕРФЕЙС, ГРАФИК, ПЛОСКОСТЬ,

Целью работы является разработка графического интерфейса для нейросетевой системы обработки радиосигналов.

В процессе работы был проведён обзор и анализ средств разработки и библиотек, были определены технические и программные средства.

В результате работы была выбрана необходимая среда разработки для языка Python, изучение выбранных библиотек, реализация пользовательского интерфейса.

СОДЕРЖАНИЕ

СПИСОК СПЕЦИАЛЬНЫХ ТЕРМИНОВ И СОКРАЩЕНИЙ	6
ВВЕДЕНИЕ.....	7
1. АНАЛИЗ СРЕДСТВ РАЗРАБОТКИ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА ДЛЯ НЕЙРОСЕТЕВОЙ СИСТЕМЫ ОБРАБОТКИ РАДИОСИГНАЛОВ	8
1.1. Постановка задачи	8
1.2. Требования к разработке.....	8
1.3. Анализ средств разработки.....	8
1.3.1. Visual Studio.....	9
1.3.2. PyCharm	9
1.3.3. Spyder	10
1.4. Выбор библиотек для создания пользовательского интерфейса	11
1.5. Выводы.....	12
2. ПРИНЦИП РАБОТЫ НЕЙРОСЕТЕВОЙ СИСТЕМЫ ОБРАБОТКИ РАДИОСИГНАЛОВ	13
2.1. Описание алгоритма работы нейросетевой системы обработки радиосигналов.....	13
2.2. Описание алгоритма работы пользовательского интерфейса.....	14
2.2.1. Построение графиков по полученным данным	14
2.2.2. Отображение построенных графиков.....	14
2.2.3. Вывод оценки точности распознавания сигнала.....	14
2.2.4. Изменение параметров плоскости	14
2.2.5. Сохранение результатов.....	14
2.3. Выводы.....	16

3. РЕЗУЛЬТАТЫ РАБОТЫ ДОРАБОТАННОЙ НЕЙРОСЕТЕВОЙ СИСТЕМЫ ОБРАБОТКИ РАДИОСИГНАЛОВ	17
3.1. Организация данных.....	17
3.1.1. Организация входных данных.....	17
3.1.2. Организация выходных данных	17
3.2. Технические и программные средства	17
3.2.1. Технические средства.....	17
3.2.2. Программные средства.....	17
3.3. Результаты работа программа	17
3.4. Выводы.....	27
ЗАКЛЮЧЕНИЕ	28
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	29

СПИСОК СПЕЦИАЛЬНЫХ ТЕРМИНОВ И СОКРАЩЕНИЙ

В данном отчёте по научно-исследовательской работе применяется следующий список специальных терминов и сокращений:

Мб – мегабайт

МГц – мегагерц

ГГц - гигагерц

ОЗУ – оперативное занимающее устройство;

IDE – Integrated Development Environment – интегрированная среда разработки;

SVGA – Super Video Graphics Array – супер видео графический массив;

CSV – Comma-Separated Values – значения, разделенные запятыми

NPZ – формат файла numpy для предварительного сжатия массивов

TXT – формат текстового файла

JPG – один из растровых форматов изображения

PNG – один из растровых форматов изображения

SVG – векторный формат изображения

ВВЕДЕНИЕ

В процессе прохождения учебной практики ставится задача разработать графический интерфейс для нейросетевой системы обработки радиосигналов, приём которых был осуществлён с помощью программно-конфигурируемого радио.

Оцифрованная запись полученного на частоте 1090 МГц радиосигнала используется в системе обработки для извлечения сигналов типа АНЗН-В и создания набора данных для нейронных сетей. Эти сигналы широко применяются беспилотными летательными аппаратами для предотвращения столкновений с воздушными судами, так как технология АЗН-В предполагает трансляцию их местоположения, высоты и скорости. [5]

В системе обработки используются две конфигурации нейронной сети, одна из которых может выполнять поиск и отличие нужного радиосигнала в потоке от шумов и неполных сигналов, а другая – получать набор бит декодированного сигнала. Для упрощения работы инженерам с данными необходимо обеспечить их визуализацию, обработку, выдачу результатов, их хранение и простоту взаимодействия с системой.

1. АНАЛИЗ СРЕДСТВ РАЗРАБОТКИ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА ДЛЯ НЕЙРОСЕТЕВОЙ СИСТЕМЫ ОБРАБОТКИ РАДИОСИГНАЛОВ

1.1. Постановка задачи

Требуется разработать графический интерфейс для нейросетевой системы распознавания и декодирования радиосигналов.

1.2. Требования к разработке

Доработка системы с точки зрения графического интерфейса должна включать в себя следующий набор функции:

- построение графиков значений амплитуды и декодированных бит сигналов на основе набора выходных данных нейросетевых алгоритмов;
- кастомизация (возможность изменение отступов у графиков, задание интервалов осей, настройка видимости сетки) и визуализация (перемещение по плоскости, увеличение и уменьшение её масштаба);
- вывод результатов работы нейронных сетей и сохранение данных о сигналах в виде набора данных в файлы формата TXT, CSV, NPZ и в виде графиков в файлы формата JPG, PNG, SVG

1.3. Анализ средств разработки

Разработка нейросетевой системы обработки радиосигналов будет осуществляться на операционную систему Windows 10.

Для выбора наиболее подходящей среды разработки на языке Python сравним между собой существующие решения, проанализируем их преимущества и недостатки.

1.3.1. Visual Studio

Интегрированная среда разработки Visual Studio — это стартовая площадка для написания, отладки и сборки кода, а также последующей публикации приложений. Помимо стандартного редактора и отладчика, которые есть в большинстве сред IDE, Visual Studio включает в себя компиляторы, средства автозавершения кода, графические конструкторы и многие другие функции для улучшения процесса разработки. Visual Studio позволяет создавать и подключать сторонние дополнения (плагины) для расширения функциональности практически на каждом уровне, включая добавление поддержки систем контроля версий исходного кода (Subversion и VisualSourceSafe), добавление новых наборов инструментов (для редактирования и визуального проектирования кода на предметно-ориентированных языках программирования или инструментов для прочих аспектов процесса разработки программного обеспечения). [7]

Полнофункциональная IDE от Microsoft для Windows и Mac OS.

Преимущества:

- поддержка установки пакетов разработки Python;
- наличие средств отладки и запуска кода

Недостатки:

- отсутствие поддержки запуска на Linux OS;
- сложность установки и большой размер

1.3.2. PyCharm

Полноценная IDE разработанная специально для Python. PyCharm делает разработку максимально продуктивной благодаря функциям автодополнения и анализа кода, мгновенной подсветке ошибок и быстрым исправлениям. Автоматические рефакторинги помогают эффективно редактировать код, а удобная навигация позволяет мгновенно перемещаться по проекту. Умный поиск позволяет быстро перейти к любому классу, файлу или символу, а также к

нужному окну или действию IDE. Переход к вышестоящему методу, тесту, объявлению, вхождению или реализации осуществляется в одно нажатие. PyCharm предлагает большой набор инструментов из коробки: встроенный отладчик и инструмент запуска тестов, профилировщик Python, полнофункциональный встроенный терминал, инструменты для работы с базами данных. IDE интегрирована с популярными системами контроля версий, содержит встроенный SSH-терминал, поддерживает возможности удаленной разработки и удаленные интерпретаторы, а также интеграцию с Docker и Vagrant. С PyCharm вы сможете работать с ноутбуками Jupyter, запускать команды в интерактивной консоли Python, работать с библиотеками для научных вычислений и анализа данных, включая Matplotlib и NumPy. [6]

Преимущества:

- Специализированная IDE для Python;
- наличие средств отладки и запуска, бесплатной версии;
- кроссплатформенность

Недостатки: может медленно работать, для некоторых проектов необходима корректировка IDE

1.3.3. Spyder

IDE специализированная для разработки на Python в сфере data science, работающая на Windows, Mac OS и GNU/Linux, которая объединяет множество библиотек для научного использования: Matplotlib, NumPy, SciPy и IPython. Spyder написана на Python для Python и разработана учеными, инженерами и аналитиками данных для себя. Она предлагает уникальное сочетание расширенных функций редактирования, анализа, отладки и профилирования комплексного инструмента разработки с исследованием данных, интерактивным выполнением, глубокой проверкой и прекрасными возможностями визуализации научного пакета. Частично переведена на русский язык. Среда разработки на Python является кросс-платформенной и имеет открытый исходный код, а также

предназначается для научного программирования. Spyder интегрируется с рядом выдающихся пакетов из Python Scientific Stack, включая NumPy, SciPy, Matplotlib, Pandas, IPython, SymPy и Cython. Она использует инструментарий Qt для собственного графического интерфейса и поэтому доступна на всех платформах в Windows, в MacOS через MacPorts и в Linux. [8]

Преимущества: заточенная и оснащенная средствами для data science

Недостатки: Ограниченный функционал и предназначение для узкой сферы делают его менее функциональный чем другие IDE.

В Таблице 1 приведено сравнение основных сред разработки на Python.

Таблица 1 – Сравнение существующих средств разработки

	Visual Studio	PyCharm	Spyder
Возможность отладки и запуска	+	+	+
Кроссплатформенность	-	+	+
Наличие бесплатной версии	+	+	+
Удобство	+ / -	+	+ / -
Функционал	+	+	-

Проанализировав существующие среды разработки, было принято решение использовать IDE PyCharm.

1.4.Выбор библиотек для создания пользовательского интерфейса

Изначально нейросетевая система обработки радиосигналов разрабатывалась на языке Python 3.7.12. Поэтому и графический интерфейс для системы также будет разрабатываться на языке Python 3.7.12.

Основные преимущества языка программирования Python: [2]

- читаемость кода и простота в освоении;
- активно развивается;
- портирован и работает почти на всех известных платформах

Для разработки пользовательского интерфейса на языке Python 3.7.12 будут использоваться следующие библиотеки:

- библиотека PyQt5 разработана для языка программирования Python и предоставляет широкий функционал, а также включая в себя отдельный дизайнер для построения графического интерфейса; [3]
- библиотека Matplotlib является основной для построения научных графиков в Python, включает функции для создания высококачественных визуализаций: линейных диаграмм, гистограмм и т.д. [4]
- стандартные библиотеки Python

1.5.Выводы

Был выполнен обзор и анализ средств разработки, их преимущества и недостатки. Выбор среды разработки был сделан в пользу кросс-платформенной и многофункциональной IDE PyCharm. Также был проведён краткий обзор библиотек для разработки пользовательского интерфейса PyQt и Matplotlib,

2. ПРИНЦИП РАБОТЫ НЕЙРОСЕТЕВОЙ СИСТЕМЫ ОБРАБОТКИ РАДИОСИГНАЛОВ

2.1. Описание алгоритма работы нейросетевой системы обработки радиосигналов

Алгоритм состоит из следующих этапов: загрузка пакетов (шум или неполный пакет), проверка нейронной сети с помощью загрузки потоков, получение предсказаний нейронной сети, оценка точности распознавания сигнала, загрузка конфигурации нейронной сети

Блок-схема алгоритма работы нейросетевой системы обработки радиосигналов показана на рисунке 1.



Рисунок 1 – Блок-схема алгоритма работы нейросетевой системы обработки радиосигналов

2.2.Описание алгоритма работы пользовательского интерфейса

Алгоритм состоит из следующих этапов: построение графиков по полученным на основе предсказания нейронной сети данным, отображение построенных графиков, вывод оценки точности распознавания сигнала, изменение параметров плоскости, сохранение результатов.

2.2.1. Построение графиков по полученным данным

Построение графиков начала, амплитуды и декодированных битов сигнала, присвоение имён заголовкам осей и графиков.

2.2.2. Отображение построенных графиков

В зависимости от выбора «распознавание» или «декодирование» будут отображаться разные графики. В первом случае график начала сигнала: ось X – время, ось Y - амплитуда, во втором случае графики амплитуды и декодированных бит сигнала: ось X – время, ось Y – амплитуда.

2.2.3. Вывод оценки точности распознавания сигнала

В формы для результатов работы выводятся значения «точность» и «потери». В специальной панели управления можно изменить число знаков после запятой (по умолчанию 3).

2.2.4. Изменение параметров плоскости

Изменения происходят с графиками в зависимости от выбора параметров «распознавание» или «декодирование». При выборе «декодирование» на холсте отображаются 2 графика, параметры плоскости которых можно регулировать отдельно. Изменению поддаются интервалы осей X и Y, а также видимость сетки на каждой из осей. Помимо плоскости корректировке также поддаются и отступы в независимости от выбранного графика. Можно перемещаться по плоскости и изменять масштаб плоскости.

2.2.5. Сохранение результатов

Данные можно сохранить в виде графиков, выбрав формат JPG, PNG, SVG или в виде наборов данных (массивов), выбрав формат TXT, CSV, NPZ. В

зависимости от выбора параметра «распознавание» или «декодирование» будут сохраняться данные об определённых графиках или массивах.

Блок-схема алгоритма работы пользовательского интерфейса показана на рисунке 2.

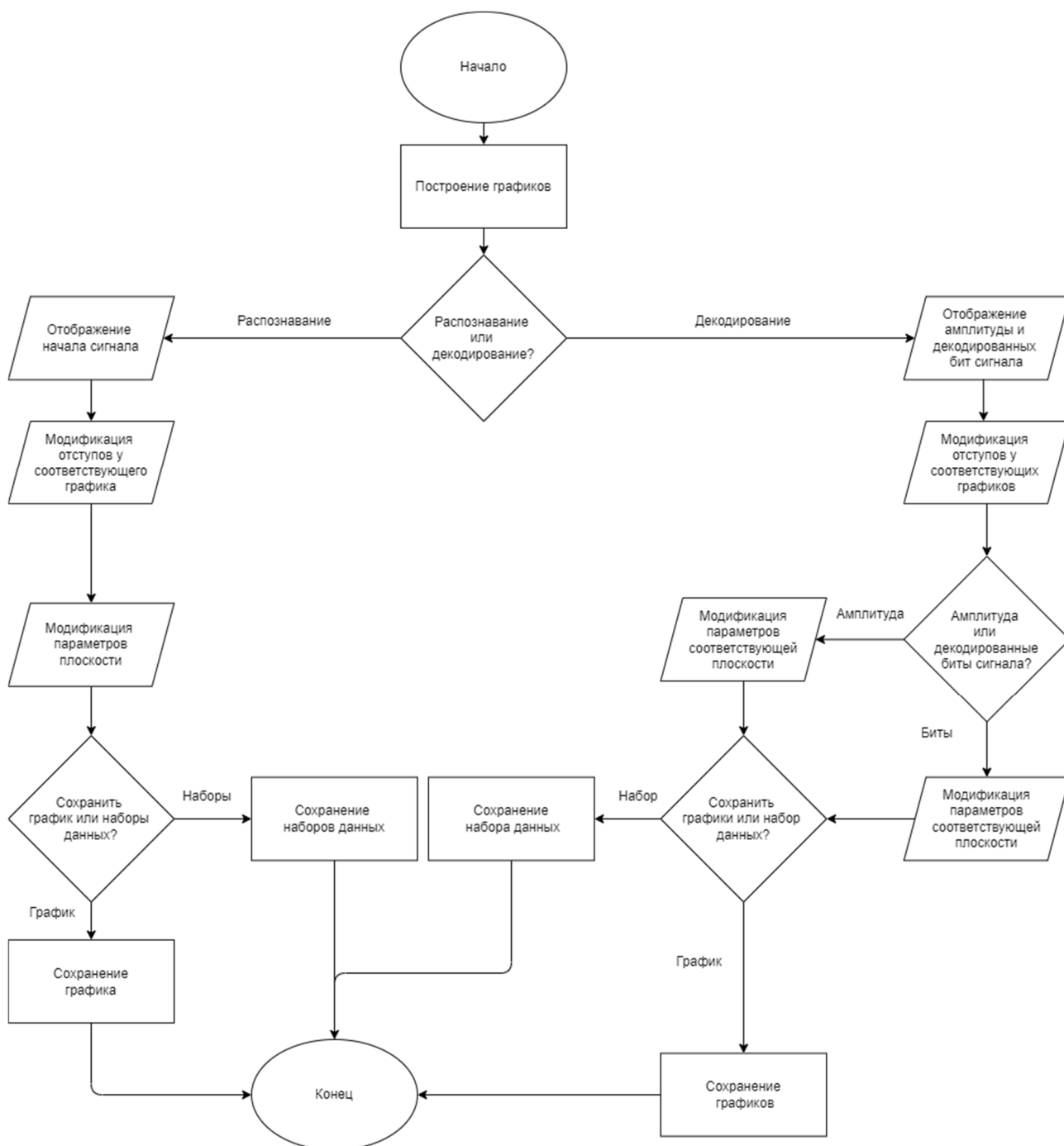


Рисунок 2 – Блок-схема алгоритма работы пользовательского интерфейса

2.3.Выводы

Был разработан алгоритм работы пользовательского интерфейса для нейросетевой системы обработки радиосигналов: были определены входные и выходные данные, параметры, которые должны будут поддаваться изменению пользователю и данные, которые необходимо визуализировать.

3. РЕЗУЛЬТАТЫ РАБОТЫ ДОРАБОТАННОЙ НЕЙРОСЕТЕВОЙ СИСТЕМЫ ОБРАБОТКИ РАДИОСИГНАЛОВ

3.1. Организация данных

3.1.1. Организация входных данных

Входными данными в программе NNSystemForProcessingRadioSignals являются файлы формата NPZ.

3.1.2. Организация выходных данных

Выходными данными в программе являются графические и текстовые файлы, сообщения в заголовках, а также графики.

3.2. Технические и программные средства

3.2.1. Технические средства

Для обеспечения функционирования программы необходим компьютер со следующими минимальными параметрами:

- процессор x64 с тактовой частотой 1,4 ГГц;
- объем ОЗУ 512 Мб;
- SVGA дисплей с минимальным разрешением 800x600;
- видеокарта;
- клавиатура;
- манипулятор типа «мышь» или другое указательное устройство

3.2.2. Программные средства

Для обеспечения работы программы компьютер должен работать под операционной системой Windows 10.

3.3. Результаты работа программа

3.3.1. Требования к программе

Программа должна обладать достаточной степенью надежности, удобства визуальной части, тестируемостью и доступностью последующих исправлений.

3.3.2. Скриншоты работы программы

В данной части продемонстрированы скриншоты работы разных частей программы.

На рисунке 3 представлено основное окно нейросетевой системы обработки радиосигналов. В нём происходит выбор действия (распознавание или декодирование сигнала), отображается график начала сигнала, панель управления плоскостью, а также значения точности и потери результатов работы.

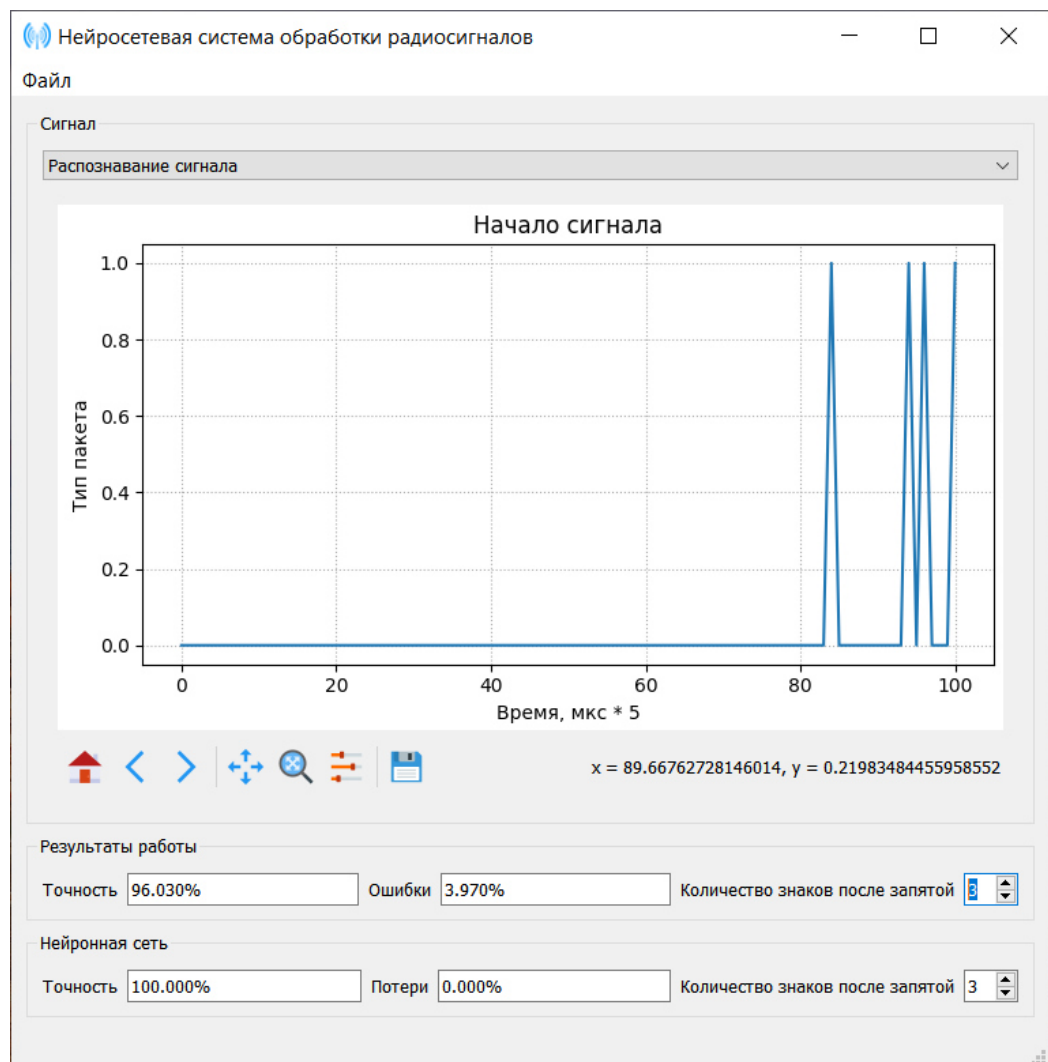


Рисунок 3 – основное окно нейросетевой системы обработки радиосигналов с графиком начала сигнала

На рисунке 4 представлено основное окно нейросетевой системы обработки радиосигналов с графиками амплитуды и декодированными битами сигнала.

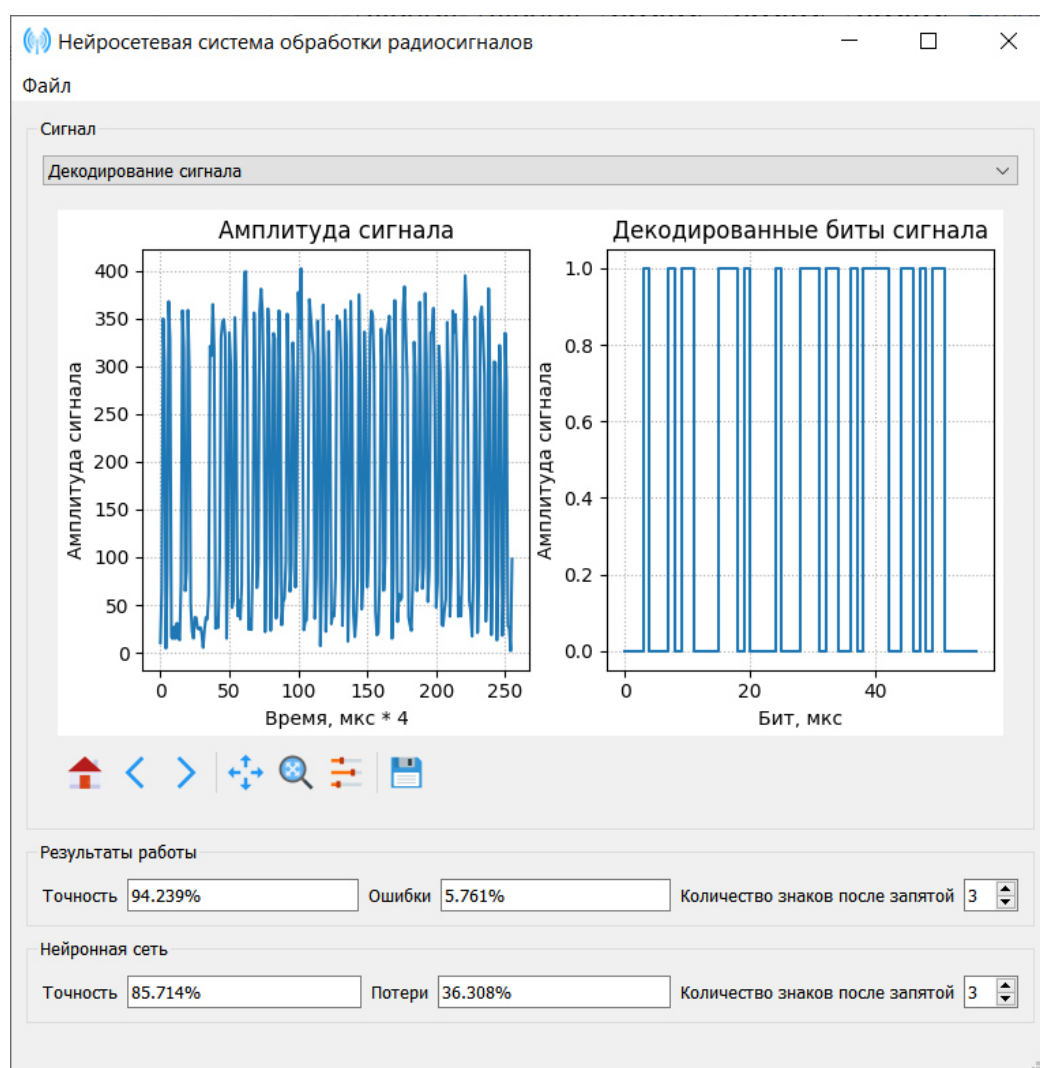


Рисунок 4 – основное окно нейросетевой системы обработки радиосигналов с графиком амплитуды и декодированными битами сигнала

На рисунке 5 продемонстрировано перемещение графика.

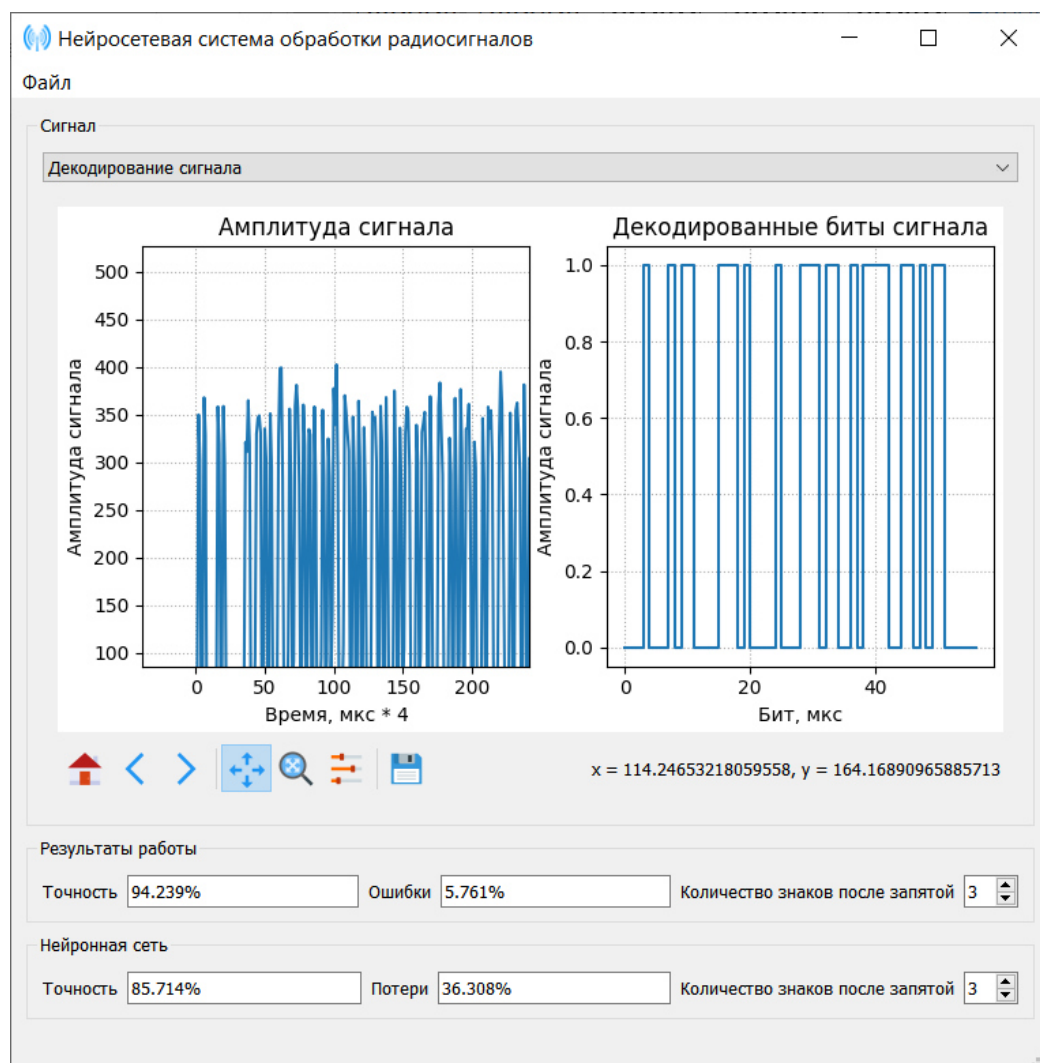


Рисунок 5 – перемещение графика

На рисунке 6 продемонстрировано изменение формата отображения результатов работы.

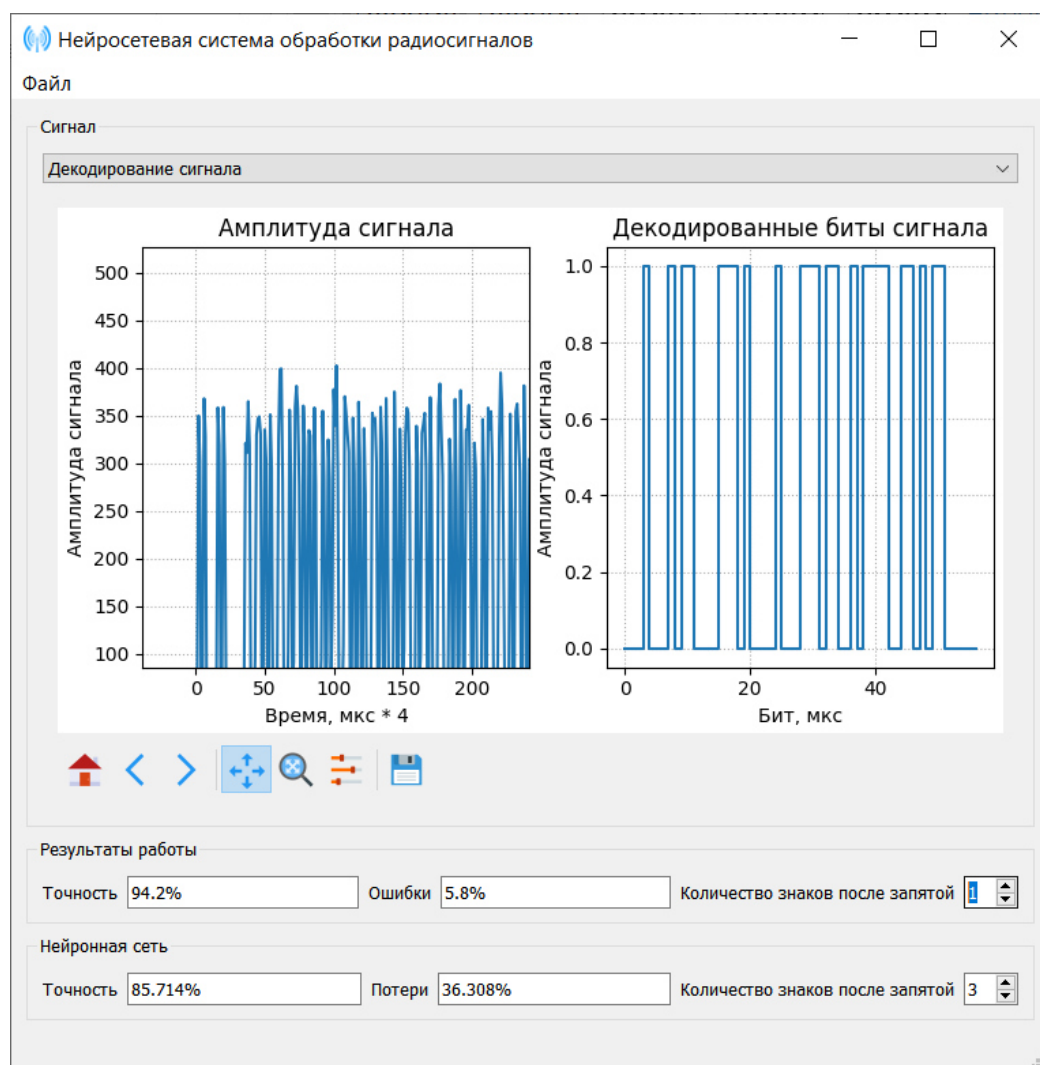


Рисунок 6 – изменение количества отображаемых знаков после запятой в формах результатов работы

На рисунке 7 продемонстрировано изменение масштаба плоскости.

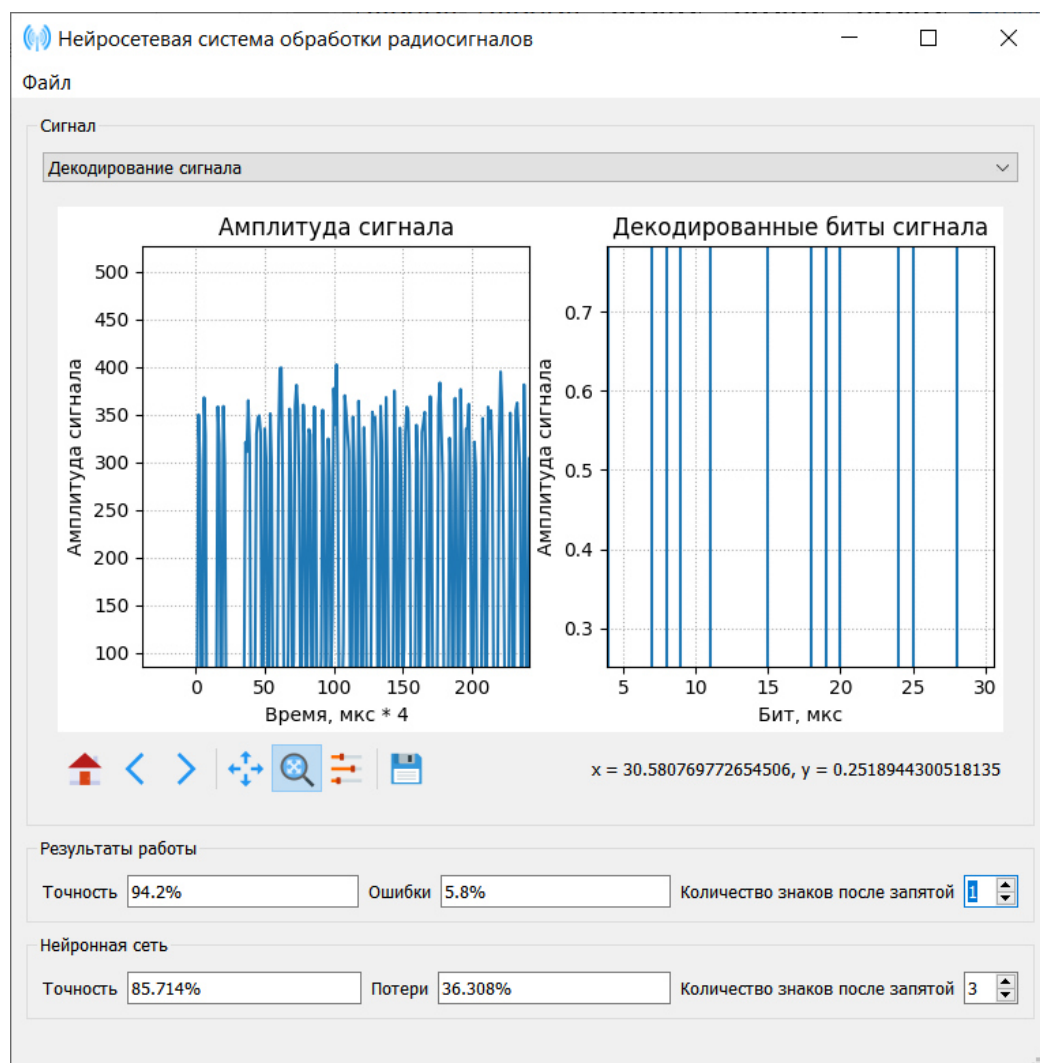


Рисунок 7 – изменение масштаба плоскости

На рисунке 8 представлено диалоговое окно настройки отступов и параметров плоскости. Оно позволяет изменить значение отступов от каждой из сторон, задать интервалы по осям X и Y, а также настроить видимость сетки для соответствующего графика.

На рисунке 9 продемонстрировано открытие файла с набором данных потока формата NPZ

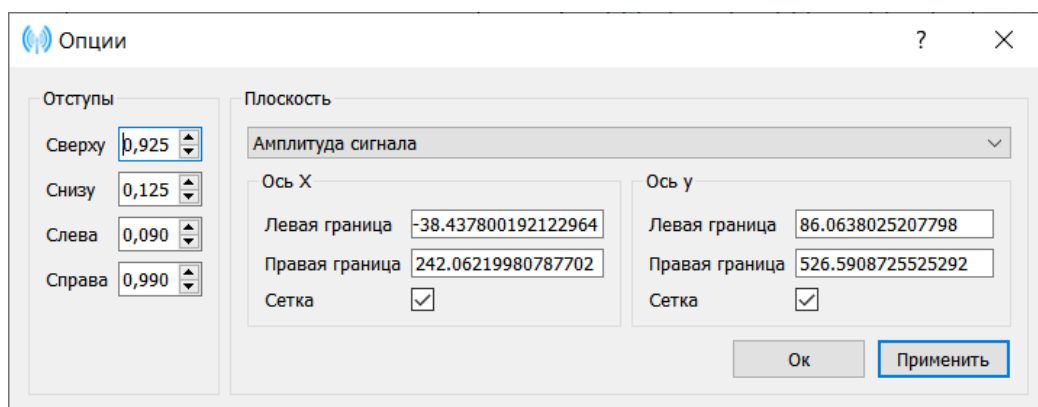


Рисунок 8 – диалоговое окно настройки отступов и плоскости

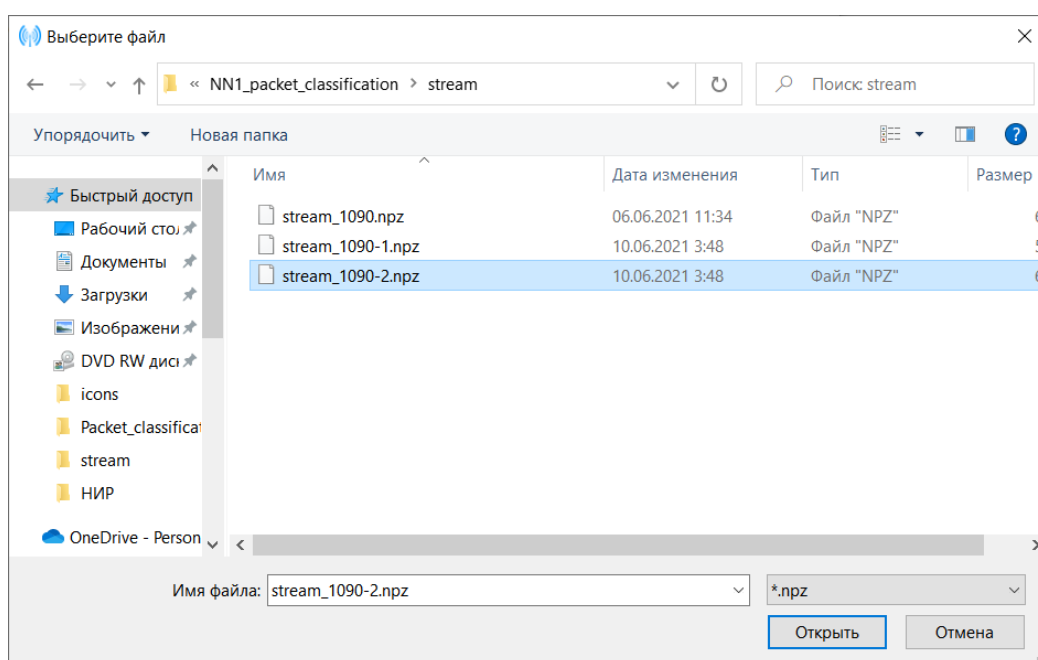


Рисунок 9 – диалоговое окно выбора файла с набором данных потока

На рисунке 10 представлено главное окно нейросетевой системы обработки радиосигналов с графиком начала сигнала, распознанным из данных набора потока файла, выбранным пользователем.

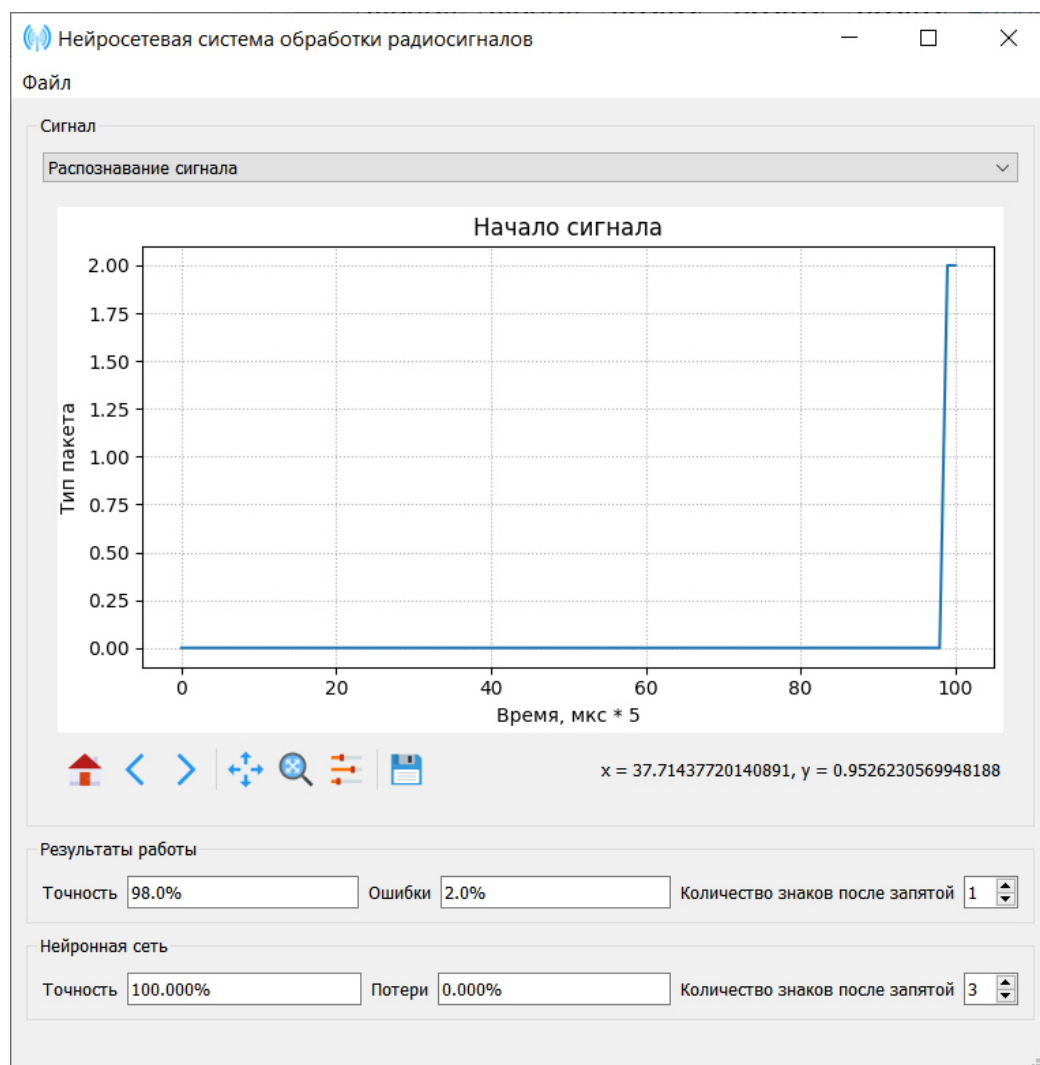


Рисунок 10 – основное окно нейросетевой системы обработки радиосигналов с графиком начала сигнала распознанным из данных набора потока файла, выбранным пользователем

На рисунке 11 продемонстрировано сохранение графика в файл формата JPG.

На рисунке 12 представлены графики в файле формата JPG.

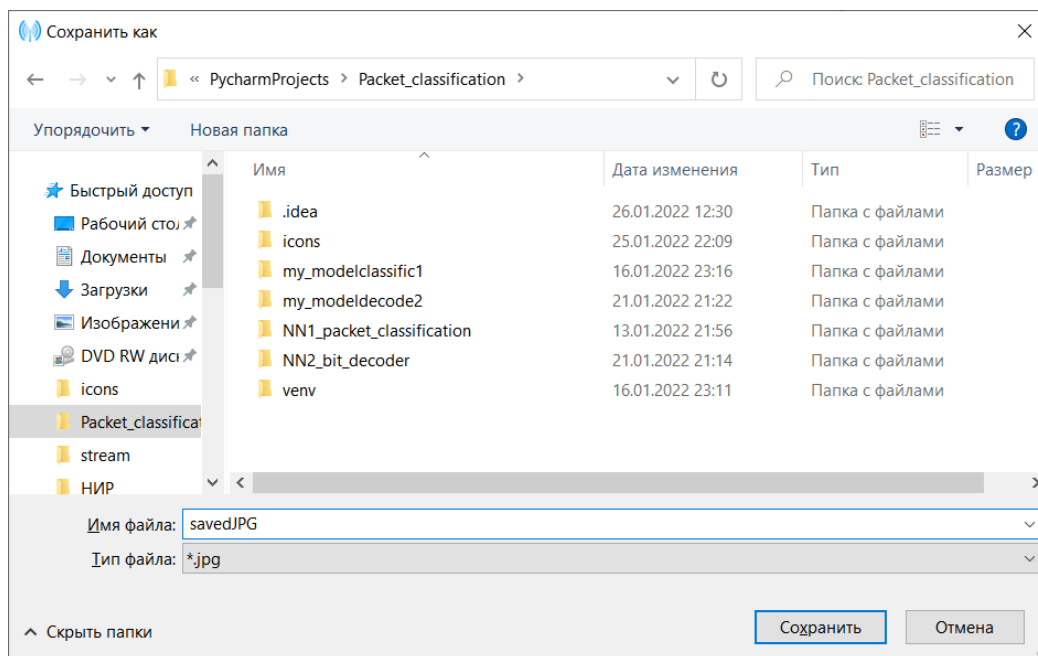


Рисунок 11 – диалоговое окно сохранения графика в файл формата JPG

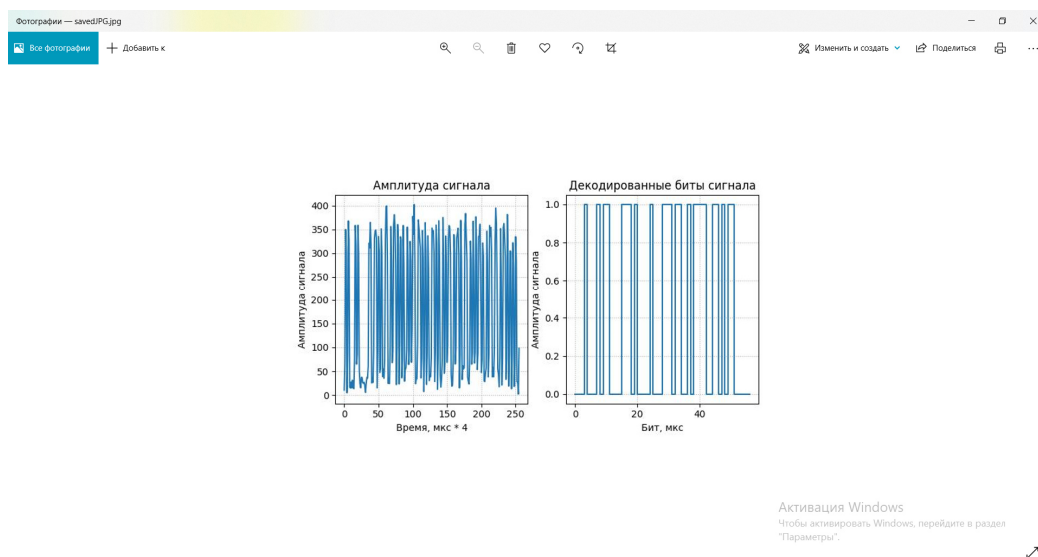


Рисунок 12 – графики в файле формата JPG

На рисунке 13 продемонстрировано сохранение графика в файл формата TXT.

На рисунке 14 представлены данные о декодированных битах в файле формата TXT.

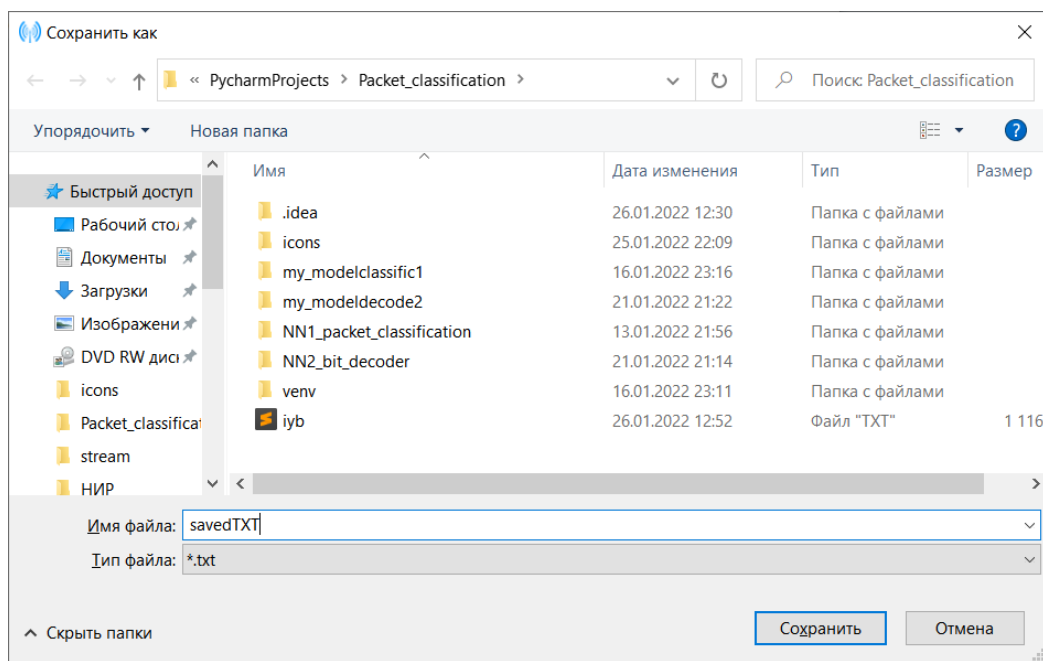


Рисунок 13 – диалоговое окно сохранения данных о декодированных битах в файл с расширением TXT

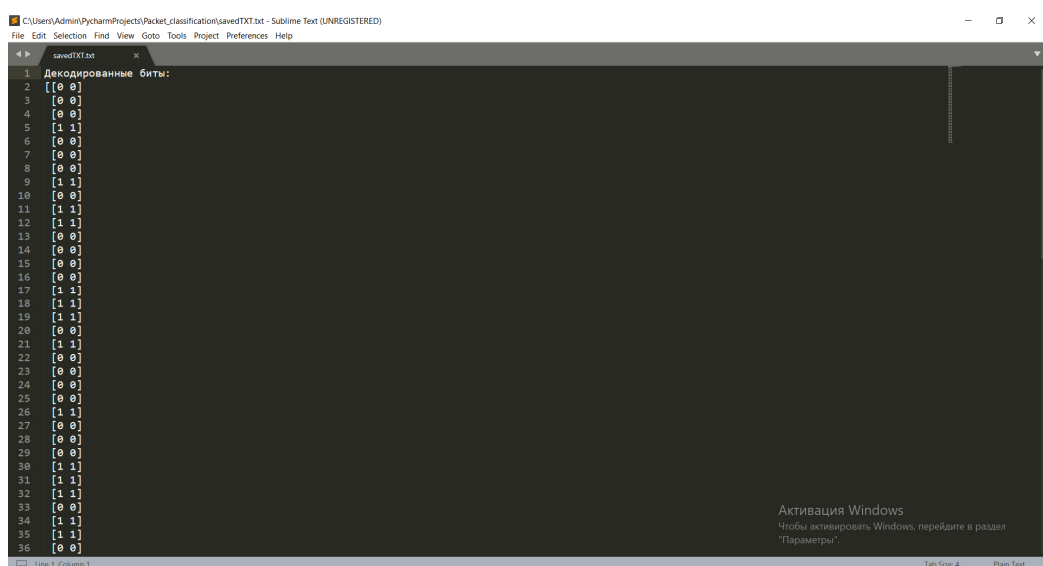


Рисунок 14 – данные о декодированных битах в файле формата TXT

3.4.Выводы

В результате выполнения программы были успешно работоспособность всех заявленных функций. Программа работает корректно: считывает данные с файлов, обрабатывает входные данные, реагирует на нажатие различных кнопок, выдаёт нужные результаты, сохраняет файлы по указанном пути.

ЗАКЛЮЧЕНИЕ

В результате прохождения проектно-технологической практики был сделан выбор средств разработки для создания пользовательского интерфейса в пользу кроссплатформенной и многофункциональной IDE PyCharm, а также были изучены такие библиотеки, как PyQt и Matplotlib.

Согласно поставленной задаче был разработан пользовательский интерфейс для нейросетевой системы обработки радиосигналов.

Согласно разработанному алгоритму были определены входные и выходные, модифицируемые и визуализированные данные.

Программа `NNSystemForProcessingRadioSignals` способна выполнять поставленные в ходе разработки задачи.

Дальнейшие перспективы развития пользовательского интерфейса:

- Модификация внешнего вида кривых (цвет, толщина, стиль);
- Кастомизация сетки (толщина, стиль, вспомогательная сетка)
- Выделение начала сигнала прямоугольником

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Лутц М. «Изучаем Python», 4-ое издание – Пер. с англ. – СПб.: Символ-Плюс, 2011. –1280 с.
2. Сайт о программировании Metanit: [Электронный ресурс]. URL: <https://metanit.com>. (Дата обращения: 28.10.2021).
3. Документация по Qt: [Электронный ресурс]. URL: <https://doc.qt.io>. (Дата обращения 11.11.2021).
4. Документация по Matplotlib: [Электронный ресурс]. URL: <https://matplotlib.org/stable/index.html> (Дата обращения 25.11.2021).
5. Возможности развития технологий ADS-B и MLAT в России. Вестник науки и образования Северо-Запада России. - 2015. - No2.: [Электронный ресурс]. URL: <https://cyberleninka.ru/article/n/vozmozhnosti-razvitiya-tehnologiy-ads-b-i-mlat-v-rossii> (Дата обращения 10.10.2021)
6. Документация по PyCharm: [Электронный ресурс] URL: <https://www.jetbrains.com/pycharm/> (Дата обращения 14.11.2021)
7. Документация по Visual Studio: <https://visualstudio.microsoft.com/> [Электронный ресурс] URL: (Дата обращения 14.11.2021)
8. Документация по Spyder: [Электронный ресурс] URL: <https://www.spyder-ide.org/> (Дата обращения 14.11.2021)