

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ
КАЗАХСТАН**

**Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ ИМЕНИ
ГУМАРБЕКА ДАУКЕЕВА»**

Кафедра Телекоммуникаций и инновационных технологий

«Допущен к защите»

Зав.кафедрой Кадылбеккызы Э.К., доктор PhD, доц.
(Ф.И.О., ученая степень, звание)
«___» _____ 2021 г.
(подпись)

ДИПЛОМНЫЙ ПРОЕКТ

На тему: Реализация технологии распознавания речи с использованием передовых архитектур ИНС

Специальность 5B071900 – Радиотехника, электроника и телекоммуникации

Выполнила Салимов Р.К. **Группа** РЭТ(ИКТ)-17-4

Научный руководитель Байкенов А.С. к.т.н., профессор
(Ф.И.О., ученая степень, звание)

Консультанты

по технической части:

Семенякин Н. В PhD ,доц.

(Ф.И.О., ученая степень, звание)

«___» _____ 2021 г.
(подпись)

по экономической части:

Габелашвили К.Р.доц

(Ф.И.О., ученая степень, звание)

«___» _____ 2021 г.
(подпись)

по безопасности жизнедеятельности:

Бекбасаров Ш.Ш., д.т.н

(Ф.И.О., ученая степень, звание)

«___» _____ 2021 г.
(подпись)

по применению вычислительной техники:

Семенякин Н. В PhD ,доц.

(Ф.И.О., ученая степень, звание)

«___» _____ 2021 г.
(подпись)

Нормоконтролер:

(Ф.И.О., ученая степень, звание)

«___» _____ 2021 г.
(подпись)

Рецензент:

(Ф.И.О., ученая степень, звание)

«___» _____ 2021 г.
(подпись)

Алматы 2021 г.

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ
КАЗАХСТАН**

**Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ ИМЕНИ
ГУМАРБЕКА ДАУКЕЕВА»**

**Институт Телекоммуникаций и космической инженерии
Кафедра Телекоммуникаций и инновационных технологий
Специальность 5B071900 – Радиотехника, электроника и телекоммуникаций**

ЗАДАНИЕ

на выполнение дипломного проекта

Студенту	Салимов Р.К.
Тема проекта	Реализация технологии распознавания речи с использованием передовых архитектур ИНС

Утверждена приказом ректора № 217 от « 27 » 10 2020 г.

Срок сдачи законченного проекта «25» мая 2021 г.

Исходные данные к проекту требуемые параметры результатов проектирования (исследования) и исходные данные объекта:

Реализация технологии распознавания речи с использованием передовых архитектур ИНС. Для моделирования работы по сегментации речи и работе сети Кохонена в программе MatLab. Реализация технологии распознавания речи с использованием передовых архитектур ИНС на языке программирования Python.

Перечень вопросов, подлежащих разработке в дипломном проекте, или краткое содержание дипломного проекта:

- 1 Нейронные сети в распознавании речи
- 2 Обучение нейронной сети
- 3 Сегментация речи
- 4 Моделирование работы нейронной сети Кохонена в программе MATLAB
- 5 Реализация распознавания речи
- 6 Безопасность жизнедеятельности
- 7 Бизнес план.

Перечень графического материала (с точным указанием обязательных чертежей):

1. Блок-схема системы распознавания речи
2. Простая нейронная сеть
3. Нейронная сеть для языкового моделирования
4. Пример многослойной сети
5. Ввод звука
6. Формантно-голосовая модель синтеза
8. Сегментация речевого сигнала
9. Сегментация зашумленного речевого сигнала
10. Успешная сегментация зашумленного речевого сигнала
11. Simulink-модель системы сегментации речи.
12. Спектрограмма сегментации речи в Simulink
13. Соревновательный слой Кохонена в MatLab
14. Распределение трех кластеров на плоскости
15. Визуализация числа векторов, ассоциированных с нейронами двумерной SOM
16. Листинг кода Python для распознавания речи
17. Схема расположения кондиционера в производственном помещении
18. План эвакуации и расположение огнетушителей на этаже
19. Диаграмма структуры расходов

Основная рекомендуемая литература:

- 1 История возникновения нейронных сетей URL – <https://neuronus.com/history/5-istoriya-nejronnykh-setej.html> (дата обращения 20.05.2021)
- 2 Начальник П. "Речевые Команды: общедоступный набор данных для распознавания речи однословного", 2017.
- 3 Доклад Томаса Зенкель. Языковое моделирование на основе классов с помощью нейронных сетей [пер. с нем.], Кáрлсруэ 2015 г;
- 4 Теуво М. Кохонен. Самоорганизующиеся карты. Издательство БИНОМ. Лаборатория знаний: 2014 С.: 659 ISBN: 978-5-99-63-1348-8

Консультанты по проекту с указанием относящихся к ним разделов проекта

Раздел	Консультант	Сроки	Подпись
Техническая часть	Семенякин Н. В PhD ,доц.	17.06.2021 г.	
Экономическая часть	Габелашвили К.Р., доцент, к.э.н.	11.06.2021	
Безопасность жизнедеятельности	Бекбасаров Ш.Ш., старший.преподаватель, д.т.н	10.04. 2020	
Применение вычислительной техники	Семенякин Н. В PhD ,доц.	17.06.2021 г.	
Нормоконтроль	Кожабаева И.Б., старший преподаватель	22.06.2021	

График

подготовки дипломного проекта

№ п/п	Наименование разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю	Примечание
1	Введение	30.01-15.02. 2021 г.	
2	Теоретическая часть. Нейронные сети в распознавании речи	21.02-28.04. 2021 г.	
3	Расчетная часть	17.05-17.06. 2021 г.	
4	Безопасность жизнедеятельности	9.04-10.04. 2020 г.	
5	Бизнес план	10.05-11.06. 2021 г.	
6	Подготовка к презентации	31.05-10.06. 2021 г.	

Дата выдачи задания

«__1__»__февраля____2021 г.

Заведующий кафедрой

(подпись) (Темырканова Э.К.)
(Ф.И.О.)

Научный руководитель проекта

(подпись) (Семенякин Н.В.)
(Ф.И.О.)

Задание принял к исполнению
студент

(подпись) (Салимов Р.К.)
(Ф.И.О.)

Аннотация

В данной дипломной работе рассматривается внедрение технологии распознавания речи с использованием усовершенствованных архитектур искусственных нейронных сетей. Во вводной части представлено описание технологии распознавания речи с использованием искусственной нейронной сети, анализируются методы распознавания речи.

В основной части исследуются искусственные нейронные сети, вычисляется сегментация речи и моделируется работа сети Кохонена. Также приводится реализация технологии на языке программирования Python. В разделе Безопасность жизнедеятельности рассматриваются вопросы системы кондиционирования и пожарной безопасности. В экономической части рассчитываются сроки окупаемости проекта, получения прибыли. Начисляются амортизационные расходы. Будет разработан Бизнес-план.

Андатпа

Бұл дипломдық жұмыста жасанды нейрондық желілердің жетілдірілген архитектураларын қолдана отырып, сөйлеуді тану технологиясын енгізуді қарастырылады. Кіріспе бөлімде жасанды нейрондық желісінді қолдана отырып, сөйлеуді танудың шолу келтірілген, сөйлеуді тану әдістері талданады.

Негізгі бөлімінде жасанды нейрондық желілер зерттеледі, сөйлеудің сегментациясы есептеледі және Кохонен желісінің жұмысын модельдейді. Python бағдарламалау тілінде технологияны енгізу де берілген. Өміртіршілік қауіпсіздігі бөлімінде кондиционерлеу жүйесі және өрт қауіпсіздігі мәселелері қарастырылады. Экономикалық бөлімінде жобаның өтелу, пайда келтіру мерзімілер есептеледі. Амортизациялық шығындар есептеледі. Бизнес-жоспар әзірленеді.

Annotation

This thesis discusses the implementation of speech recognition technology using advanced architectures of artificial neural networks. In the introductory part, a description of the speech recognition technology using an artificial neural network is presented. speech recognition methods are analyzed.

In the main part, artificial neural networks are studied, speech segmentation is calculated, and the operation of the Kohonen network is modeled. The implementation of the technology in the Python programming language is also provided. In the section Life Safety, the issues of air conditioning and fire safety are considered. In the economic part, the payback period of the project and profit generation are calculated. Depreciation expenses are accrued. A business plan will be developed.

Содержание

Введение	8
1 Нейронные сети в распознавании речи.....	9
1.1 Речь как канал связи	9
1.2 Индивидуальные особенности речи	9
1.3 Идентификация и верификация по голосу	9
1.4 Принцип работы распознавания речи	10
1.5 Языковая модель N-граммы	12
1.6 Нейронные сети в языковом моделировании	14
1.7 Оценка языковой модели.....	16
1.8 Архитектура нейронной сети	17
1.9 Выходной слой с классами	20
1.10 Подведение итогов.....	21
1.11 Обучение нейронной сети	22
1.12 Функции ошибок.....	22
2 Расчетная часть	25
2.1 Ввод звука	25
2.2 Наложение первичных признаков на вход нейросети	27
2.3 Модель синтеза речи.....	27
2.4 Сегментация речи	29
2.5 Моделирование работы нейронной сети Кохонена в программе MATLAB	37
2.6 Реализация распознавания речи	44
3 Безопасность жизнедеятельности	46
3.1 Анализ условий труда вычислительного центра.....	46
3.2 Расчет системы кондиционирования воздуха	46
3.3 Разработка эвакуационных путей	50
3.4 Расчет количества и типа огнетушителей	53
4 Финансовый план	57
4.1 Расчет инвестиционных затрат	57
4.2 Расчет эксплуатационных расходов	58
4.3 Амортизационные отчисления, материальные затраты	62
4.5 Доходы предприятия	64
4.6 Расчет экономического эффекта, эффективной и срока окупаемости..	64
4.7 Расчёт результатов от создания и использования ПО	67
4.8 Расчёт основных показателей экономической эффективности.....	68
Заключение.....	71
Список литературы	72

Введение

Исследования в области нейронных сетей начались в 40-х годах XX века. Первое систематическое исследование искусственных нейронных сетей было предпринято Маккалоком и Питтсом в 1943 году. Они предложили компьютерную модель нейронной сети, разработанную на основе теории функционирования мозга и математических алгоритмов.

Основными задачами, с которыми сталкиваются нейронные сети, являются задачи распознавания образов. Цель состоит в том, чтобы классифицировать входное изображение. На первом этапе подаются справочные изображения, принадлежность которых к определенному классу уже известна заранее. Затем нейронная сеть анализирует входы с помощью определенного алгоритма и коррелирует их с эталонным методом. Другими словами, нейронная сеть выполняет классификацию изображений. Кластеризация объектов также происходит при распознавании образов. Основное различие между задачей кластеризации и задачей классификации заключается в том, что объекты не указаны, к какому классу они принадлежат. Алгоритм разбивает весь набор объектов на кластеры, объединяя объекты, похожие по признакам. Кластерный анализ широко используется в области исследований рынка ценных бумаг, прогнозирования цен на акции и решения многих других классов задач.

Все больше и больше приложений взаимодействия между людьми и техническими устройствами зависят от статистической языковой модели, из-за все более активной роли электронных систем в нашей повседневной жизни. Например, языковые модели вносят важный вклад в перевод текста на другой язык, классификацию и транскрибирование звукозаписи и видеозаписи, а также в распознавание рукописных текстов.

Искусственные нейронные сети - один из наиболее эффективных способов решения проблем распознавания. Распознавание речи используется для биометрических целей безопасности для идентификации голоса конкретного человека. Рынок распознавания речи и распознавания речи является одним из самых быстрорастущих рынков в языковой индустрии.

1 Нейронные сети в распознавании речи

1.1 Речь как канал связи

Язык - один из видов коммуникативной деятельности человека; канал коммуникации для получения информации из общечеловеческого социального и исторического опыта. Язык существует устно и письменно. Устная составляющая выражается в произнесенных и воспринимаемых высказываниях, составленных с помощью звуков (акустического кода). В образовании этих компонентов принимают участие полость рта и носа, гортань, язык, губы и челюсть – голосовой аппарат человека.

1.2 Индивидуальные особенности речи

Устная речь индивидуальна по звучанию. Индивидуальные особенности речи-совокупность особенностей, определяемых особенностями строения голосовых органов. Определяющими характеристиками являются тембр голоса и темп речи. Кроме того, индивидуальные особенности формируются размером речевого тракта человека, размером голосовых связок и их мышц, объемом легких, характеристикой грудных мышц, речевыми способностями, особыми признаками, языковыми дефектами и др. Человек может не только осознать смысл сказанного, но и на подсознательном уровне в процессе общения идентифицировать говорящего, различать голоса других людей. Все факторы, влияющие на распознавание речи, учитываются и могут быть использованы для выделения индивидуальных особенностей голоса каждого человека.

1.3 Идентификация и верификация по голосу

Процесс распознавания личности говорящего голосом можно разделить на два основных подпроцесса: идентификацию и верификацию. Идентификация-процесс распознавания пользователя голосом, путем парного сравнения полученного образца с голосовыми моделями, уже имеющимися в базе данных. Результатом является фонограмма, которая, скорее всего, соответствует образцу ввода. Этот процесс предполагает наличие достаточно большой базы данных.

Верификация-процесс установления принадлежности полученного образца и языкового шаблона, хранящегося в базе данных, к одному и тому же голосу. Результатом является подтверждение или опровержение системой идентификатора, представленного пользователем. Этот процесс не включает в себя большую базу данных: его можно рассматривать как частный случай процесса идентификации, если в базе данных есть только один языковой шаблон.

1.4 Принцип работы распознавания речи

Задача распознавания речи - анализировать разговорную речь, а затем чаще всего переводить ее в написанный текст. При этом система распознавания речи, используемая для этого, может быть разделена на различные компоненты. Как показано на рисунке 1.1, голосовой сигнал сначала предварительно обрабатывается. Затем он декодируется с помощью информации акустической модели и языковой модели.

Акустическая модель есть такая функция, которая принимает на вход некоторые признаки в небольшом участке акустического сигнала (фрейме) , потом выдает распределение вероятностей разных фонем на этом самом фрейме. Так акустическая модель может дать звуку восстановить, что было произнесено или сказано с той или другой степенью уверенности.

Одной из целей языковой модели является выявление правил в языке и, таким образом, определение вероятности данных сочетаний слов. Эта дополнительная информация может быть отброшена, среди прочего, маловероятными гипотезами, выдвинутыми другими компонентами приложения.

Среди всех возможных последовательностей слов словаря W ищут наиболее вероятную последовательность слов W^* , заданную акустическая (звуковым) сигналом A . Это можно изменить с помощью теоремы Байеса на фундаментальную формулу распознавания речи [3]:

$$W^* = \arg \max \frac{P(W) \cdot P\left(\frac{A}{W}\right)}{P(A)} \quad (1.1)$$

где $P(A|W)$ — вероятность (также условная), которую мы получаем из наших данных за нее отвечает акустическая модель;

$P(W)$ — вероятность события A (основная, априорная оценка) за нее отвечает языковая модель;

а $P(A)$ — константа нормировки, которая ограничивает вероятность значением, за нее отвечает декодер.

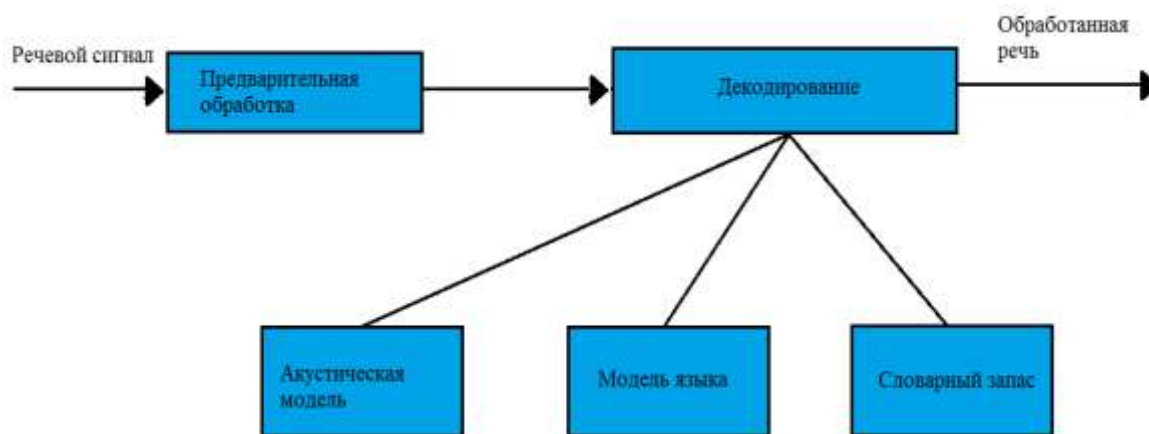


Рисунок 1.1 – Блок-схема системы распознавания речи

Языковая модель вычисляет вероятность $P(W)$, а плотность вероятности $p(A|W)$ определяется акустической моделью. Значение языковой модели проявляется прежде всего у омофонов, то есть у слов, обладающих одним и тем же произношением, но различным значением. Однако с помощью грамматики и за счет дополнительной информации из контекста разговора человеку легко понять, что первое предложение имеет значительно больше смысла. При этом омофоны являются лишь одним из примеров того, как языковая модель может улучшить качество системы распознавания речи. Даже при нечетко произнесенных словах, высоком уровне шума и во всех ситуациях, когда акустическая модель не может однозначно определить слово, языковая модель предоставляет важную информацию о правдоподобности возможных последовательностей слов.

В прошлом были исследованы различные предложения для создания надежных языковых моделей. При этом предложения можно грубо разделить на детерминированные и вероятностные языковые модели. В детерминированной модели языка определенная последовательность слов разрешена или запрещена. С этой целью, например, с помощью формальной грамматики можно определить, какие последовательности слов разрешены в языке, который нужно распознать. Это особенно подходит для очень ограниченной системы диалога, где человеку задаются вопросы, на которые мало ответов. Например, при автоматической покупке билета по телефону формальные грамматики поэтому обладают своими сильными.

С другой стороны, при свободном языке вероятностные языковые модели оказались более значимыми, поскольку человек не всегда придерживается правильной грамматики языка. Кроме того, определение всех грамматических правил естественного языка в модели языка представляет собой большую проблему. Таким образом, в вероятностных языковых моделях последовательности слов присваивается вероятность. Это, как правило, оценивается на основе тренировочного текста. При этом, как правило, оценивается частота появления последовательности слов, ее вероятность.

Однако по мере увеличения вычислительной мощности в наши дни можно также изучить вероятность последовательности слов для подмножества словарного запаса языка за приемлемое время с помощью нейронной сети. При этом нейронные сети представляют собой семейство алгоритмов статистического обучения, вдохновленных человеческим мозгом. Нейронные сети не применяются в таких областях, как распознавание рукописного текста, распознавания образов и при прогнозировании курсов акций, в которых стандартные алгоритмы достигнут в основном состояния сопоставимых результатов.

Поскольку рассмотрение всего словаря при вычислении вероятностей с помощью нейронной сети слишком трудоемко, здесь необходимо использовать другие методы. Таким образом, в этой работе слова словаря делятся на классы слов. При этом изучаются различные методы с помощью нейронных сетей с помощью классов слов, чтобы определить вероятности различных последовательностей слов [3]..

1.5 Языковая модель N-граммы

Чтобы оценить вероятность последовательности слов, существует два основных предложения. Во-первых, из грамматики можно извлечь модель языка. Однако этот принцип знает последовательности слов, которые вполне могут быть в разговорах и которые не допускаются только из-за грамматики языка, с вероятностью 0. Кроме того, грамматика может быть только искусно смоделирована. Поэтому в современных системах распознавания речи в значительной степени используются статистические языковые модели. Может быть при заданной последовательности слов w_1, \dots, w_n вероятность $P(w_1, \dots, w_n)$ последовательности вычисляется следующим образом с помощью условных вероятностей:

$$P(w_1, \dots, w_n) = P(w_1) \cdot \prod_{i=2}^n P\left(\frac{w_i}{w_1}, \dots, w_{i-1}\right) \quad (1.2)$$

При этом вероятность слова $P(w|h)$ зависит от контекста h предшествующих ему слов. В результате при более большом контексте количество вычисляемых вероятностей увеличивается экспоненциально. Для вполне обычного предложения с 20 Вт мест и словарным размером 300 000 слов необходимо предварительно подсчитать 300000^{20} вероятностей. Поскольку это нереализуемо, в N-граммовой языковой модели длина контекста слова ограничивается N-1. Следовательно, образуются эквивалентные классы $c(h)$. Таким образом, делается предположение, что вероятность слова зависит только от предшествующих N-1 слов.

Следовательно, для вероятности последовательности слов получается:

$$P(w_1, \dots, w_n) = P(w_1) \cdot \prod_{i=2}^n P(w_i / w_{i-N+1}, \dots, w_{i-1}) \approx \prod_{i=1}^n (w_i | c(h)) \quad (1.3)$$

N-грамм с $N=1$ называется униграммой, для биграмм- $N=2$, а для триграмм- $N=3$. N-граммы с более длинным контекстом не имеют специального названия.

Теперь, чтобы оценить вышеупомянутые вероятности, выбирается учебный текст. Затем в этом тексте отмечается, как часто слово w встречается в эквивалентном классе его контекста $c(h)$. Ниже $\#$ обозначает количество появления следующей последовательности слов в обучающем тексте. Таким образом, искомая вероятность вычисляется с помощью N-граммовой языковой модели с уравнением [3]:

$$P_{nGram}(w|c(h)) = \frac{\#c(h)w}{\#c(h)} \quad (1.4)$$

При описанном подходе вероятность N-грамм, которые не встречаются в учебном тексте, приближается к 0. Этого нельзя избежать с помощью этой процедуры, так как может быть выбран только один, наконец, большой учебный корпус, и, следовательно, не все слова будут происходить во всех возможных контекстах. Чтобы присвоить значимую вероятность еще не увиденному N-грамму, ниже представлены различные методы сглаживания.

Простой подход заключается в том, чтобы предположить, что каждый N-грамм встречается в тренировочном теле по крайней мере α -раз:

$$P_{Laplace}(wi|c(h)) = \frac{\#c(h)wi + \alpha}{\#c(h) + \alpha|V|} \quad (1.5)$$

Благодаря этому распределению тривиально вероятность для каждого слова больше 0, однако добавляется относительно большая масса вероятности, поскольку на слово словаря α искусно создаются новые примеры. Это может сильно исказить распределение вероятностей, особенно при большом словарном запасе.

Чтобы избежать этого, можно перераспределить только фиксированную долю вероятности $0 < \beta < 1$:

$$P(wi|c(h))_{\beta} = (1 - \beta) \frac{\#c(h)wi}{\#c(h)} + \beta \frac{1}{N} \quad (1.6)$$

В этой формуле N обозначает количество слов словаря. Но теперь, как и в случае с лапласовым, все слова, еще не увиденные в контексте, сопоставляются с той же вероятностью. Чтобы избежать этой проблемы, можно использовать информацию из N-грамм более низкого порядка.

Это реализуется при линейной интерполяции, как описано в:

$$P_{interp} = \lambda P(wi|w_{i-n+1} \dots w_{i-1}) + (1 - \lambda) P_{interp}(wi|w_{i-n+2} \dots w_{i-1}) \quad (1.7)$$

где P_{interp} — вероятность слова;

λ — коэффициент масштабирования, зависящий от $w_{i-n+1} \dots w_{i-1}$.

Однако здесь все чаще прибегают к менее точной информации N-грамм более низкого порядка. Чтобы избежать этого, сглаживание отката

возвращается к модели более низкого порядка только для невиданных N-Gramme.

$$P_{backoff} = \begin{cases} P(w_{i-n+1} \cdots w_{i-1}), & \text{wenn } w_{i-n+1} \cdots w_{i-1} > 0 \\ \gamma(w_{i-n+1} \cdots w_{i-1})P_{backoff}(w_{i-n-2} \cdots w_{i-1}), & \text{son} \end{cases} \quad (1.8)$$

При этом P^* обозначает слегка пониженное распределение, а γ -коэффициент масштабирования, так что вероятности складываются в единицу [3].

1.6 Нейронные сети в языковом моделировании

Нейронной сетью обычно называют семейство алгоритмов статистического обучения, вдохновленных человеческим мозгом. В нем многие взаимосвязанные нейроны образуют нейронную сеть. В нейронную сеть по нейронам, зависящим от весов их связей, течет информация. Эти веса изучаются на основе тренировочного тела. Простая нейронная сеть, сгруппированная в входной слой, скрытый слой и выходной слой, показана на рисунке 1.2. Круги при этом представляют собой нейроны, стрелки между нейронами означают, что эти нейроны связаны друг с другом. Поскольку в этой нейронной сети нейроны каждого слоя соединены с нейронами следующего слоя, называется нейронной сетью прямой связи.

По мере роста вычислительной мощности в наши дни становится возможным "изучать" языковую модель с помощью нейронной сети. При таком подходе сначала каждому слову словаря присваивается m -мерный вектор слова $v \in \mathbb{R}^m$. После этого функция вероятности, которая сопоставляет вероятности со словами в зависимости от их контекста, печатается с векторами характеристик слов контекста. Как векторы признаков, так и параметры функции истинности - вероятности изучаются нейронной сетью.

Эта концепция представлена на рисунке 1.3. Три слова w_{i-3} , w_{i-2} и w_{i-1} сначала отображаются на векторах характеристик, уже изученных ранее для каждого слова. Затем они образуют вход для нейронной сети. Нейронная сеть обучается таким образом, что нейроны выходного слоя изучают функцию $P(w_i | w_{i-3}w_{i-2}w_{i-1})$.

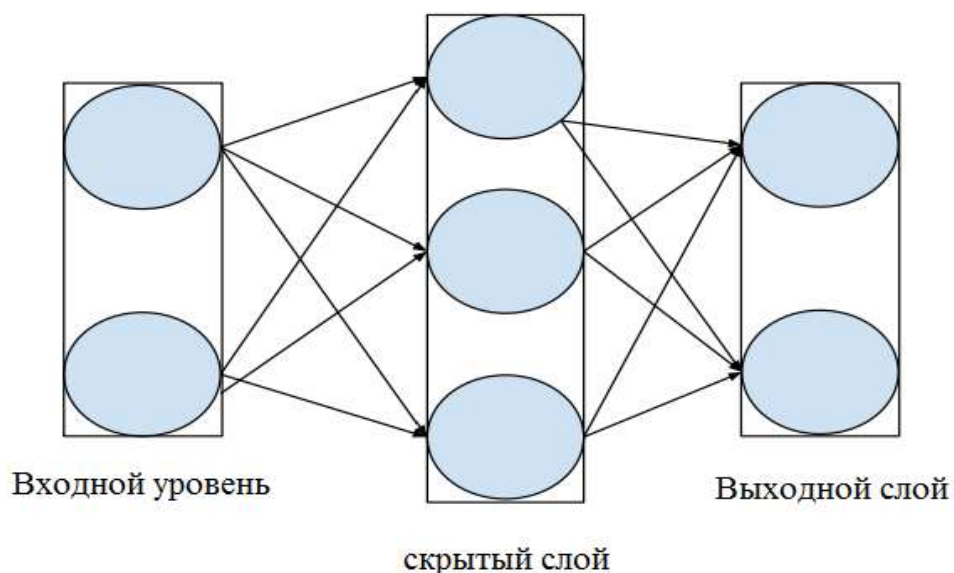


Рисунок 1.2 – Простая нейронная сеть

Одним из преимуществ этого подхода является то, что для аналогичных слов обычно генерируются аналогичные векторы характеристик. Таким образом, при гладкой функции вероятности одно и то же слово в аналогичном, но различном контексте также сопоставляется с аналогичной вероятностью. Таким образом, появление слова в контексте увеличивает как его вероятность в этом контексте, так и вероятность этого слова в аналогичном контексте.

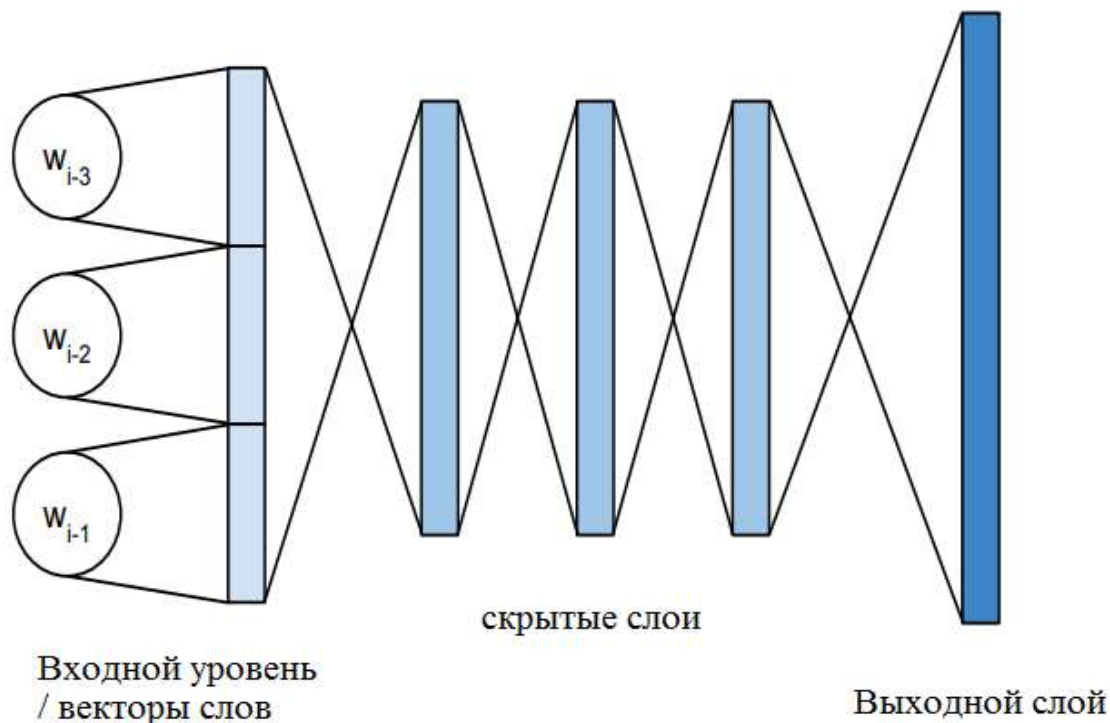


Рисунок 1.3 – Нейронная сеть для языкового моделирования

1.7 Оценка языковой модели

В этом разделе теперь представлены методы для оценки языковой модели.

Метод определения частоты ошибок. Очевидным способом является включение новой языковой модели в существующую систему распознавания речи и сравнение результатов. Для оценки системы распознавания речи это, как правило, определяет гипотезу на записанном аудиосигнале. Затем эта гипотеза сравнивается со ссылкой, ранее созданной человеком, путем определения частоты ошибок слов (англ. word error rate). Для этого вычисляется наименьшее количество заменителей V , удалений L и вставок E слов, чтобы перейти от гипотезы к ссылке. Затем результат делится на количество слов ссылки N :

$$WER = \frac{V+L+E}{N}, \quad (1.9)$$

где V – количество заменителей слов;

L – количество удалений слов;

E – количество вставок слов.

Недостатком метода частоты ошибок слов является то, что при этом оценивается не только модель языка. Например, если акустическая модель уже хорошо распознает разговорную запись, то частота ошибок слов двух разных языковых моделей будет абсолютно не так сильно отличаться, как если бы из-за плохой акустической модели было много возможностей улучшения.

С другой стороны, преимущество этого критерия оценки заключается в том, что языковая модель тестируется непосредственно в области практического применения и может быть измерена тем, улучшает ли улучшенная языковая модель качество системы распознавания речи[3].

Метод недоумения. Недоумение - это мультипликативная величина, обратная вероятности, присвоенной набору тестов языковой моделью, нормированной на количество слов в наборе тестов. Это другой подход к оценке качества языковой модели заключается в вычислении извращенности или нелогичности модели. Она описывает среднюю степень ветвления языковой модели. Для тестовой записи с N словами и языковой моделью, определяющей вероятность появления слова, учитывая его контекст $P(w_i / w_1 \dots w_{i-1})$, оно определяется следующим образом:

$$\sqrt[N]{\prod_{i=1}^N \frac{1}{P(w_i | w_1 \dots w_{i-1})}} \quad (1.10)$$

Таким образом, языковая модель, которая присваивает десяти словам одинаковую вероятность 0.1 независимо от их контекста, имеет извращение

10 для каждой тестовой записи любой длины, существующей из этих W . Однако, если в тестовой записи есть два слова с вероятностью $1/3$, а остальные восемь слов с вероятностью $1/24$, то запись имеет извращение $\sqrt[3]{3 \cdot 3 \cdot 24} = 6$. Таким образом, языковая модель, которая лучше описывает фактическое распределение вероятностей слов текста, также приписывает ему меньшее извращение.

Энтропия определяется логарифмом бин-арена извращения. Эта мера описывает среднее количество бит на слово, необходимое для оптимального двоичного кодирования текста, точно соответствующего распределению вероятностей $P(w_i | w_1 \dots w_{i-1})$ языковой модели.

1.8 Архитектура нейронной сети

Нейронные сети делятся на синхронные и асинхронные. В синхронных нейронных сетях в каждый момент времени свое состояние меняет лишь один нейрон. В асинхронных – состояние меняется сразу у целой группы нейронов, как правило, у всего слоя. Можно выделить две базовые архитектуры – слоистые и полносвязные сети. Ключевым в слоистых сетях является понятие слоя.

Слой – один или несколько нейронов, на входы которых подается один и тот же общий сигнал.

Слоистые нейронные сети – нейронные сети, в которых нейроны разбиты на отдельные группы (слои) так, что обработка информации осуществляется послойно.

В слоистых сетях нейроны i -го слоя получают входные сигналы, преобразуют их и через точки ветвления передают нейронам $(i+1)$ слоя. И так до k -го слоя, который выдает выходные сигналы для интерпретатора и пользователя. Число нейронов в каждом слое не связано с количеством нейронов в других слоях, может быть произвольным.

В рамках одного слоя данные обрабатываются параллельно, а в масштабах всей сети обработка ведется последовательно – от слоя к слою. Однако сигнал не всегда подается на все нейроны слоя. В когнитроне, например, каждый нейрон текущего слоя получает сигналы только от близких ему нейронов предыдущего слоя. Слоистые сети, пример изображен на рисунке 1.4, в свою очередь, могут быть однослойными и многослойными. В многослойной сети первый слой называется входным, последующие – внутренними или скрытыми, последний слой – выходным. Таким образом, промежуточные слои – это все слои в многослойной нейронной сети, кроме входного и выходного. Входной слой сети реализует связь с входными данными, выходной – с выходными. Таким образом, нейроны могут быть входными, и скрытыми.

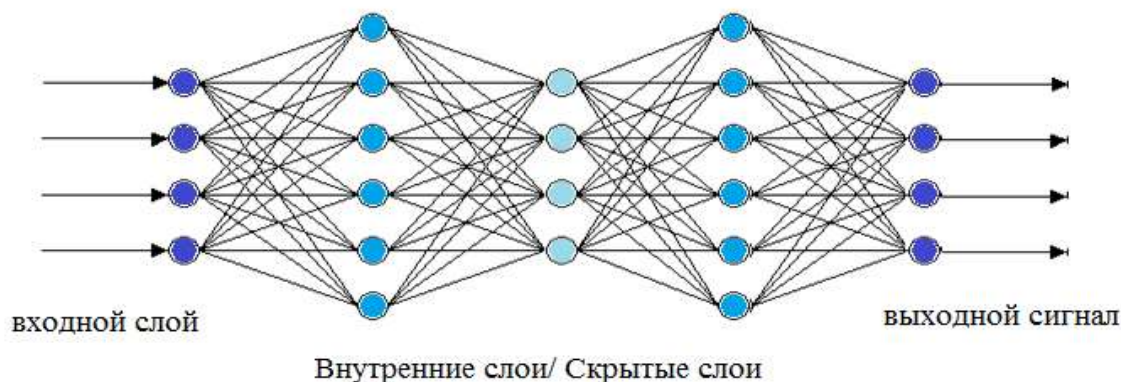


Рисунок 1.4 – Пример многослойной сети

Нейронная сеть, используемая в этой работе для языковой модели, состоит из входного слоя, нескольких скрытых слоев и выходного слоя. При этом, каждый нейрон одного слоя связан с каждым нейроном следующего слоя, можно также говорить о нейронной сети прямая связь.

Ниже более подробно описана архитектура нейронной сети на основе используемых слоев. При этом в выходном слое, с помощью которого можно вычислить апостериорные вероятности возникновения слова словаря, рассматриваются отдельные слова словаря, классы слов или подклассы. Таким образом, помимо структуры выходного слоя, рассматриваются также возникающие возможности для расчета вероятности появления одного слова.

Каждый нейрон входного уровня нейронной сети ожидает вещественное число в качестве входных данных. Таким образом, в качестве входных данных рассматривается не индекс слова, а вектор характеристики соответствующего слова. Для доступа к векторам характеристик используется так называемая кодировка 1 of n. Для этого векторы характеристик хранятся по строкам в матрице. В качестве входных данных для нейронной сети вместо слова используется строка, в которой хранится вектор характеристики слова. Затем соответствующие векторы характеристик выбираются и проецируются на входной слой[3].

Входной уровень. Теперь входной уровень зависит от длины контекста $|h|$, таким образом, $|h|$ векторы характеристик. Следовательно, величина входного уровня зависит от длины рассматриваемого контекста и от измерения векторов признаков. Таким образом, входной слой состоит из $d \cdot |h|$ нейронов. В качестве активирующей функции для нейронов следует использовать постоянно дифференцируемую функцию, производная которой не является постоянной 0. Таким образом, функция знака не может быть и речи. Поскольку при обновлении весов нейронной сети используется производная функции активации, при знаковой функции веса не изменялись. Кроме того, в качестве функции активации должна использоваться нелинейная функция, иначе нейронная сеть могла бы представлять только линейную функцию. Функция, используемая во входном слое, - это функция сигмоиды:

$$\text{sig}(x) = \frac{1}{1+1^{-x}} \quad (1.11)$$

Функция генерирует значения от 0 до 1 для каждого нейрона. На рисунке 1.5 показаны две описанные функции активации.

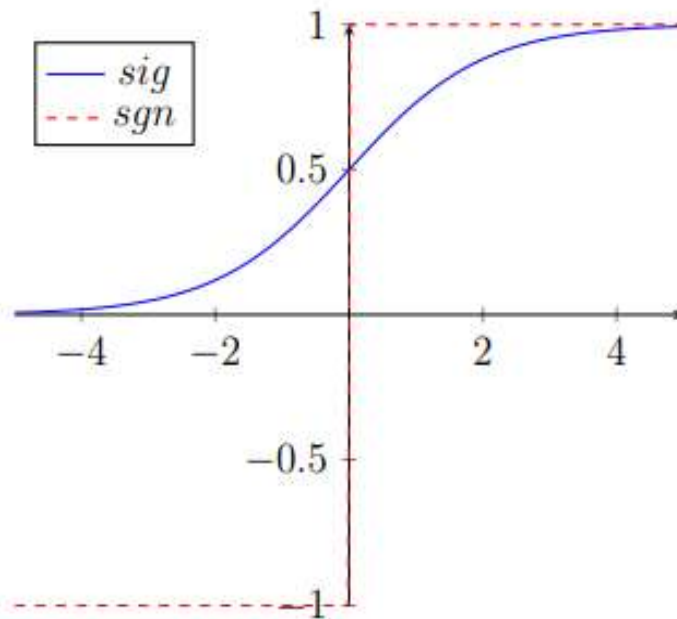


Рисунок 1.5 – Возможные функции активации: Сигмоида sig и функция знака sgn

Скрытые слои. Затем за входным слоем следуют несколько скрытых слоев. Без этих слоев также невозможно было бы аппроксимировать нелинейные функции. Нейронная сеть прямой связи уже может приближаться ко всем возможным функциям со скрытым слоем, при условии, что скрытый слой обладает достаточным количеством нейронов. Скрытые слои, как и входной слой, используют функцию сигмоиды в качестве функции активации.

Вход скрытого слоя зависит от вывода предыдущего слоя $a_{i-1} \in \mathbb{R}^m$, весов между нейронами двух слоев $W \times \mathbb{R}^{m \times n}$ и смещения $b \times \mathbb{R}^n$. При этом m обозначает количество нейронов предыдущего слоя, а n – количество нейронов только что рассматриваемого слоя:

$$a = \text{sig}(a_{i-1}^T \cdot W + b) \quad (1.12)$$

При этой сигмоиде компонентно применяется к элементам вектора.

Выходной слой. Задача выходного уровня – вычислить вероятности появления тех или иных слов. Чтобы реализовать это, каждый нейрон этого слоя репрезентирует одно слово или класс слов. В качестве функции активации используется функция Softmax:

$$\text{softmax}(x)_i = \frac{e^{x_i}}{\sum_{n=1}^N e^{x_n}} \quad (1.13)$$

Здесь N - количество нейронов в выходном слое. Эта функция присваивает каждому нейрону вероятность от 0 до 1. Кроме того, из-за нормирования в этой функции возникает функция вероятности, поскольку все вероятности складываются в 1.

Эквивалент скрытых слоев здесь теперь вычисляет выход в зависимости от вывода a последнего скрытого слоя, весов W и смещения b между двумя слоями:

$$P(w_i|h) = \text{softmax}(a^T \cdot W + b) \quad (1.14)$$

В базовой реализации с нейросетью рассматривалось только подмножество словарного запаса. Список слов этого подмножества обозначается термином Shortlist ниже. Таким образом, выход сети вычисляет для всех рассмотренных слов $w \in \text{Shortlist}$ вероятность появления $P(w|h)$ слова, учитывая его контекст h . Вероятности слов шорт-листа при этом интерполируются с помощью N -граммовой языковой модели P_B с параметром a_s . Поскольку языковая модель нейронной сети P_N определяет только вероятность подмножества слов словаря, при расчете вывода также требуется нормализация. Следовательно, следующая формула вычисляет вероятность вывода слова шорт

$$P(w|h) = \begin{cases} P_N(w|h) \cdot a_s \cdot P_s + (1 - a_s) \cdot P(w|h) & \text{fall } w \in \text{Shortlist} \\ P_B(w|h) & \text{sont} \end{cases} \quad (1.15)$$

Ниже теперь представлены различные возможности для просмотра вероятностей с нейронной сетью для всех мест большого словаря, не занимая слишком много вычислительного времени для оценки языковой модели.

1.9 Выходной слой с классами

Один из подходов для этого-разделить еще не рассмотренные слова на классы. Ниже $c(w)$ обозначает функцию, которая сопоставляет класс со словом w . Теперь нейронная сеть, помимо слов шорт-листа, также рассматривает классы c_1, \dots, c_n . Для этого выходной уровень увеличивается на количество классов. Другой способ - использовать свой собственный выходной уровень для классов.

Вероятность возникновения затем определяется как для классов $P(c_i|h)$, так и для слов шорт-листа, данного контексту h с нейронной сетью. Для определения этой вероятности для слов, разделенных на классы, дополнительно используется вероятность слова $P(w|c(w))$, заданного его классу. Например, эта вероятность может быть проксимирована через вероятность униграммы с помощью тренировочного текста ar следующим образом:

$$P_{Unig}(w_i|c(w)) = \frac{\#w_i}{\sum_{w \in c(w_i)} \#w} \quad (1.16)$$

Затем при вычислении вероятностей слов можно снова провести различие между языковой моделью нейронной сети и N-граммовой языковой моделью. При этом можно использовать два параметра интерполяции. Параметр as указывает вес языковой модели нейронной сети при рассмотрении слов шорт-листа. Для взвешивания вероятных остальных слов используется параметр интерполяции ar . Из-за различных параметров сумма вероятностей слов в коротком списке и оставшихся слов требуется для языковой модели N-граммы, а также для языковой модели нейронной сети. В следующей формуле PBs и PBR обозначают сумму вероятности слов шорт-листа и остальных слов для N-граммовой языковой модели. Аналогичным образом, PNs и PNr обозначают соответствующие суммированные вероятности для языковой модели нейронной сети[3]:

$$P_{Bs} = \sum_{w \in Shortlist} P_B(w), \quad (1.17)$$

$$P_{Br} = \sum_{w \in Shortlist} P_B(w) = 1 - P_{Bs}, \quad (1.18)$$

$$P_{Ns} = \sum_{w \in Shortlist} P_N(w), \quad (1.19)$$

$$P_{Nr} = \sum_{w \in Shortlist} P_N(w) = 1 - P_{Ns} \quad (1.20)$$

Таким образом, при таком подходе сумма вероятностей слов шорт-листа не изменяется по сравнению с N-граммовой языковой моделью. Таким образом, это относится[3].:

$$\sum_{w \in Shortlist} P(w|h) = \sum_{w \in Shortlist} P_B(w|h) \quad (1.21)$$

1.10 Подведение итогов

На входной уровень проецируются векторы характеристик из трех слов. Эти векторы изучаются на отдельном этапе. За входным слоем теперь следуют три скрытых слоя.

Последняя часть нейронной сети теперь доступна для расчета вероятностей и состоит, по крайней мере, из одного выходного слоя, который требует нейрона для каждой рассматриваемой вероятности. В первой модели нейронная сеть рассматривает только подмножество словаря, называемого шорт-листом. Все слова этого подмножества получают нейрон за раз в выходном слое.

Другая модель рассматривает весь словарный запас, разбивая слова на классы. Вероятность появления класса слова затем рассматривается нейронной сетью. Затем с помощью униграммной вероятности слова в этом классе можно вычислить вероятность появления слова. Однако слова шорт-листа по-прежнему рассматриваются индивидуально. В этой модели вероятности слов и классов можно рассматривать вместе в одном слое,

назначая слою еще один нейрон для каждого класса. Кроме того, можно использовать еще один выходной слой, который добавлен только для классов.

Кроме того, с помощью другого выходного уровня также можно вычислить вероятность появления слова, данного его классу. Для этого слова дополнительно разбиваются на подклассы, так что каждое слово класса сопоставляется с другим подклассом. Таким образом, подклассы заменяют вероятность униграммы внутри класса. Для этого дополнительный выходной уровень, в свою очередь, имеет нейрон для каждого подкласса [3].

1.11 Обучение нейронной сети

Существует три способа обучения искусственным нейронным сетям:

1) обучение с учителем (supervised) – контролируемое или наблюдаемое обучение. Нейронная сеть обеспечивает желаемый результат (выходы) в соответствии с входными данными;

2) обучение без учителя (unsupervised) – неконтролируемое или необнаруженное обучение. В этом случае не все данные, используемые для обучения, известны правильными ответами;

3) смешанное (гибридное) обучение – включает в себя усиленное обучение (reinforcement learning), стохастический метод (Stochastic), пакетный метод (Batch) и другие.

Обучение начинается с ввода данных обучения. При первом запуске параметры нейронов задаются произвольно. Затем вычисляется значение ошибки, возникающей в результате сравнения с ожидаемыми данными. Происходит установка параметров n , с целью уменьшения ошибки функции.

Затем после завершения обучения тестируется на основе данных обучения на специальных данных. Считается, что используются данные, которые ранее не были включены в тренировки.

Тестирование обучения отличается тем, что тестовые данные реализуют проверку точности. Эта точность может служить критерием качества конфигурации сети. Если результирующая точность не соответствует заданным критериям, следует изменить структуру нейронной сети и повторить этапы обучения [3].

1.12 Функции ошибок

Чтобы определить оптимальные веса, сначала необходимо определить меру ошибок, с помощью которой можно сравнить выходы нейронных сетей с различными весами W . Ниже используются среднее квадратичное отклонение (англ.: mean squared error) MSE и перекрестная энтропия (англ.: cross entropy) CE. Где среднее квадратичное отклонение для даты обучения (x, y) определяется следующим образом [3]:

$$MSE_{(x,y)}(W) = \frac{1}{nl} \sum_{i=1}^{nl} (y^i - O_i(x))^2 \quad (1.22)$$

С другой стороны, перекрестная энтропия определена в:

$$CE_{(x,y)}(W) = \sum_{i=1}^n -y^i \log(O_i(x)) - (1 - y^i) \log(1 - O_i(x)) \quad (1.23)$$

Таблица 1.1 - Параметры скрытых слоев

Наименование	первый слой	дальнейшие слои
corruption	0.3	0.2
Функция ошибки	MSE	CE
Скорость обучения	0,001	0,01
Размер пакета	128	128
Количество пакетов	600 000	400 000

Инициализация. Прежде чем начать фактическую тренировку нейронной сети, сначала ищут подходящие начальные значения для весов нейронной сети. При этом один за другим веса между двумя слоями выбираются таким образом, чтобы нейронная сеть могла как можно лучше реконструировать примеры обучения, которые были дополнительно шумны. При этом вход $(k + 1)$ -го слоя соответствует выходу k -го слоя. При этом изменяется определенное количество входов первого слоя, зависящее от параметра corruption. При этом corruption обычно обозначает вероятность изменения характеристики входных данных.

Поскольку обновление весов требует вычислительного времени и, следовательно, это не требуется после каждой даты тренировки, данные тренировки разбиваются на пакеты одинакового размера. Оцениваются все примеры обучения этого нейрона, а затем вычисляется ошибка с помощью функции ошибки. После этого происходит обновление весов нейронной сети. При этом заданная до этого скорость обучения указывает, насколько сильно изменяются веса, чтобы улучшить ошибку для увиденных примеров. Одним из преимуществ небольшого обновления является то, что веса обновляются более часто, а более большое обновление веса связки приводит к более значимым обновлениям веса [3].

Фаза обучения. После завершения этой фазы в нейронную сеть добавляется еще один выходной слой, и начинается фактическая фаза обучения. Теперь веса оптимизируются с помощью алгоритмов. Мысль этого алгоритма заключается в возврате ошибки из слоя в слой к весам, которые оказали влияние на ошибочный вывод. при этом эти веса затем изменяются в зависимости от параметра обучения, так что в тех же примерах обучения ошибка в будущем будет меньше. Эта процедура является частным случаем градиентной процедуры.

Обучение разделено на разные этапы, причем нейронная сеть тренируется в каждом этапе со всеми данными обучения. После каждого этапа ошибка проверяется на так называемых данных проверки, чтобы узнать, оказывают ли изменения весов положительное влияние на данные проверки.

Это позволяет избежать перенастройки весов в соответствии с данными тренировки.

В любой этап скорость обучения может быть изменена. Эта процедура начинается со скорости обучения α и сохраняет ее до тех пор, пока ошибка, измеренная в данных проверки, не уменьшится менее θ_1 . После этого скорость обучения обычно сокращается вдвое. Обучение прекращается, если ошибка уменьшается менее чем на θ_2 после снижения скорости обучения на предыдущем шаге [3].

2 Расчетная часть

2.8 Ввод звука

Ввод звука будет идти в звуковую карту или в файлы формата Microsoft Wave в кодировке РСМ (разрядностью 16 бит, частотой дискретизации в 22050 Гц). Работа с файлами предпочтительней, так как в них можно повторять процессы их обработки нейронной сетью многократно, что необходимо при обучении.

Чтобы звук можно было подать на вход нейронной сети, над ним необходимо проводить преобразования. Известно, что представление звука во временной форме не очень эффективно и не показывает характерных особенностей звукового сигнала. Спектральное представление речи дает гораздо больше информации. А для получения спектра используется набор полосовых фильтров, настроенных на выбор различных частот, или используется дискретное преобразование Фурье. Результирующий спектр затем подвергается различным преобразованиям, например изменению логарифмического масштаба (как в пространстве амплитуд, так и в пространстве частот). Могут быть приняты во внимание некоторые свойства речевого сигнала, например: понижение информативности высокочастотных участков спектра, логарифмическую чувствительность человеческого уха, и т.д.

Как правило, полностью описать речевой сигнал одним его спектром невозможно. Для получения спектральной информации вам также потребуется информация о динамике речи. Для его получения используются дельта-параметры, которые являются производными по времени основных параметров.

Таким образом, параметры принятого речевого сигнала рассматриваются как его основные свойства и представляют сигнал на дальнейших этапах обработки. Процесс ввода звуковой информации изображен на рисунке 2.1:



Рисунок 2.1 – Ввод звука

При обработке файла по нему перемещается окно ввода, размер которого равен размеру окна дискретного преобразования Фурье (ДПФ).

Смещение окна относительно предыдущего положения можно регулировать. В каждой позиции окна он заполняется данными (система работает только со звуком, каждый сэмпл кодируется в 16 битах). Если вы вводите звук в реальном времени, он будет записан блоками одинакового размера.

После ввода данных в окно перед вычислением ДПФ на него накладывается окно сглаживания Хэмминга [2].:

$$w(k) = 0,54 - 0,46 \cdot \cos\left(\frac{2\pi i}{N-1}\right), \quad (2.1)$$

где N – размер окна ДПФ

Наложение окна Хэмминга немного снижает контраст, но вы можете убрать боковые лепестки резких частот (рис. 2.2), сделав гармоническую композицию речи особенно заметной.

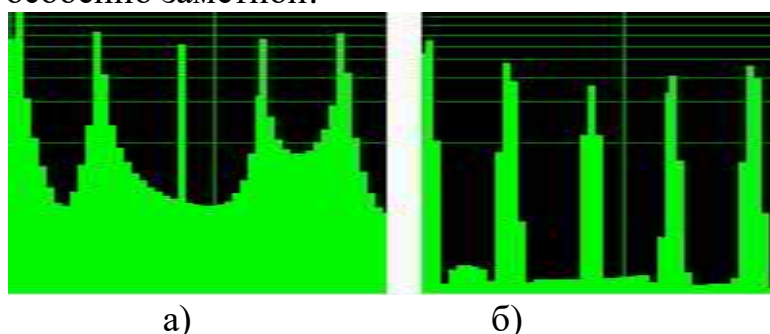


Рисунок 2.2 – Действие окна сглаживания Хэмминга (логарифмический масштаб) а) без окна сглаживания; б) с окном сглаживания Хэмминга

Затем дискретное преобразование Фурье вычисляется с использованием алгоритма быстрого преобразования Фурье. Амплитудный спектр и информация о фазе получают из результатов действительных и мнимых коэффициентов. Информация о фазе отбрасывается и рассчитывается энергетический спектр:

$$E(i) = \sqrt{\operatorname{Re} C(i) \cdot \operatorname{Im} C(i)}, \quad (2.2)$$

Из-за того что данные находящиеся в обработке не могут содержать мнимой части, то из-за свойства ДПФ результата получается симметричным, т.е. $E[i] = E[N-i]$. Вот почему, информативный размер части спектра NS равен $N/2$.

Все расчеты в нейросети производятся над числами с плавающей точкой и большая часть сигналов ограничены диапазоном $(0.0, 1.0)$, так что результирующий спектр нормализуется до 1,00. Для этого каждая из составляющих вектора делится по длине:

$$e(i) = \frac{E(i)}{|E|}, \quad (2.3)$$

$$|E| = \sum_{i=0}^{NS-1} E(i), \quad (2.4)$$

Информативность неодинакова в различных частях спектра: в области низкочастотной информации будет больше, чем в той же высокочастотной области. Из-за этого чтобы избежать ненужных затрат на входы в нейронную сеть, необходимо уменьшить количество элементов, которые принимают информацию в области с высокой степенью, или сжать эту область в области частотных пространств. И более распространен метод (из за его простоты) – логарифмическое сжатие (non-linear frequency scales) [2]:

$$m = 1125 \cdot \log(0.0016f + 1), \quad (2.5)$$

где f – частота в спектрах, Гц;

m – частота в новых сжатых частотных пространствах.

2.9 Наложение первичных признаков на вход нейросети

После нормализации и сжатия спектр накладывается на вход нейронной сети. Входы нейросети не выполняют никаких решающих функций, а только передают сигналы дальше в нейронную сеть. Выбор числа входов – сложная задача, потому что при малом размере входного вектора возможна потеря важной для распознавания информации, а при большом существенно повышается сложность вычислений (только при моделировании на РС, в реальных нейросетях это неверно, так как все элементы работают параллельно).

Благодаря большой разрешающей способности ((большому количеству) входов можно выделить гармоническую структуру речи и, следовательно, определить высоту голоса. При низкой разрешающей способности (небольшом количестве) входов возможно только определение формантной структуры.

Дальнейшее исследование этой проблемы показало, что для распознавания достаточно только информации о формантной структуре. На самом деле человек одинаково распознает нормальный голосовой язык и шепот, хотя у последнего отсутствует голосовой источник. Источник речи предоставляет дополнительную информацию в виде интонации (изменение высоты тона во время произнесения), и эта информация очень важна на самых высоких уровнях обработки речи. Однако в первом приближении можно ограничиться только получением формантной структуры, а для этого, с учетом сжатия неинформативной части спектра, достаточно количество входов выбран в диапазоне 50 ~ 100.

2.10 Модель синтеза речи

Существуют множество методов синтезов речи, но большинство из них основаны примерно на двух моделях: компилятивные синтезы – это синтез речи через объединение (составления) записанного образца отдельных звуков (речевых примитивов - фонем), произносимых говорящим, и формантно–голосовая модель, в которой голосовой тракт человека может моделироваться с разной степенью точности. Первая модель требует более кропотливой работы

для создания надежных баз данных, и изучить ее самостоятельно будет чрезвычайно сложно. Вторая модель, с другой стороны, может допускать широкий спектр самообучения, может быть идеально интегрирована в модели нейронных сетей, но из-за сложности моделирования речевого тракта людей точность синтезированных звуков может быть довольно низкой, и они разборчивы даже при относительно простом моделировании синтезированных звуков. Схематическое изображение модели синтеза речи показано на рисунке 2.3. Для построения модели мы использовали данные о артикуляционном аппарате человека, а также данные по фонетике и лингвистике [2].

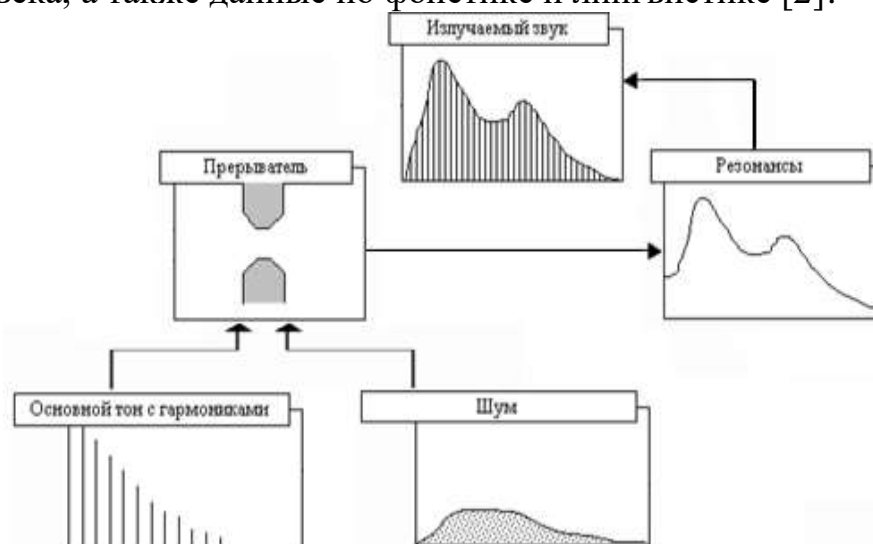


Рисунок 2.3 – Формантно-голосовая модель синтеза

Построения модели – это больше всего упрощение того, что нужно моделировать. В этом важно определить компромиссы между качествами модели (т.е. пригодность её для решений поставленных задач), и её сложности. Для этого надо выбрать более значимые параметры системы которая исследуется. В случае нашем, центральными характеристиками считаются:

- центральные частоты каждой форманты. Из-за того форманта является резонансом в тракте речи, у неё есть частота огибающей и резонанса. Вид огибающей может определяться экспериментальным способом, в первом же приближении это Гауссовы распределения;
- частоты основного тона. Определяющие параметры голосовых источников, характеризуется по высоте голосов;
- число формант. Число активных формант, участвующих в речеобразовании. Выбирается экспериментально, ориентировочно 4;
- вклад в каждую форманту. На какое количество форманта может воздействовать на основной сигнал;
- шумовые частоты. Формирование шума – является сложным процессом и может зависеть от большого количества факторов – давлении и скорость воздушных потоков, геометрических форм воздушных трактов, акустического свойства материалов – и вот почему моделирование этих

процессов на уровне физическом может представлять собой серьезные задачи и требуется построение всего человеческого речевого аппарата. Альтернативность этому – представление звуков как белые шумы, спектры которых распределены по законам (например, по Гауссу) и относительно некоторых частот. Законом распределений возможно подбирать экспериментальные, и остается у нас один переменный параметр – центральные частоты, которые намного упростят моделирование.

2.11 Сегментация речи

Сегментация речевого сигнала - является одной из важнейших задач в системах распознавания речи. Речевые сигнатуры следует читать в определенных сегментах, озвученных сегментах речевого сигнала. Форма речи характеризуется частотами формант, которые измеряются на известных гласных, скорость артикуляции рассчитывается длительностью процессов перехода между артикуляционно-акустическими участками. Для решения задачи обратной необходима сегментация - восстановление формы речевого тракта по акустическому сигналу, которое применяется в таких областях, как системы сжатия или синтезаторы речи из любого текста, передача голоса в сотовых сетях, автоматическое распознавание речи.

Сегментация в ручную может быть использовано в системах для исследования или в предварительных проектах. Но для этого требуется значительные затраты по времени и усилиям: во-первых, в непрерывной речи между словами нет пауз, во-вторых, коартикуляция, которая может происходить на границах последовательно генерируемых звуков, значительно облегчает восприятие и понимание языка. речи, но задача посложнее: найти границу сегмента. Кроме того, из-за субъективности слухового и зрительного восприятия человека сложно точно воспроизвести результат сегментации в ручную.

Проблемы такого рода не возникают при автоматической сегментации, которая также может быть подвержена ошибкам, но все же может давать воспроизводимые результаты. Существует два основных типа алгоритмов сегментации речи. Первый включает алгоритмы, выполняющие сегментацию речи при условии, что последовательности фонем в этих фразах известны. Второй же тип алгоритма априорную информацию о фразе не использует, а определяет границу сегмента по степени изменения акустических свойств сигнала. Для автоматической сегментации рекомендуется использовать только общие свойства речевого сигнала, поскольку на данный момент нет конкретной информации о содержании речи.

Для формантного метода текстозависимой проверки векторы признаков могут быть вычислены на определенных сегментах речевого сигнала. Большинство существующих систем распознавания речи используют способ, основанный на обработке речевого сигнала кадр за кадром, когда сигнал делится на пересекающиеся кадры с шагом смещения и определенной длины. Длина самого сегмента зависит от скорости изменения артикуляционного

аппарата при генерации речи, а шаг сдвига точно контролирует скорость, с которой можно отслеживать динамику изменений характера в речи. Поскольку для оценки динамики изменения спектральных составляющих сигнала используется малая длительность изображения, шаг смещения выбирается таким образом, чтобы, с одной стороны, детали изменения спектра не сглаживались, а с другой стороны, не подвергались влиянию на какие-либо существенные изменения. И вот почему, длина кадра и шаг смещения обычно объединяются в пары. В существующих системах распознавания речи используется длительность кадра, которая может составлять 20-30 мс, и, например, шаг смещения составляет 10-20 мс.

Во время покадровой обработки речевого сигнала могут возникать промежутки в спектре сигнала, которые называются антиформантами. Также существует проблема, заключающаяся в том, что большинство согласных в основном непериодические, и поэтому их спектры полностью или частично непрерывны, что отличает их от спектров гласных, которые являются периодическими. Эти недостатки могут привести к неточным значениям формант, что в точности влияет на надежность распознавания речи. Идеальным было бы вычислить векторы характеристик речи на сегментах речевого сигнала, которые соответствуют фонемам, входящим в состав слова.

Распознавание речевой активности это важная часть многих аудиосистем, таких как распознавание голоса и автоматическое распознавание речи. Обнаружение речевой активности может быть особенно трудным в ситуации с низким отношением сигнал / шум (SNR), когда речь затруднена шумом.

В данном примере используется сети краткосрочной памяти (LSTM), которые являются типом рекуррентной нейронной сети (RNN), подходящей, чтобы изучить данные времени и последовательности сигнала. Сеть LSTM может изучить зависимости между временными шагами последовательности. Слой LSTM (lstmLayer) может обнаружить временную последовательность в прямом направлении, в то время как двунаправленный слой LSTM (bilstmLayer) может обнаружить временную последовательность и в прямом и в обратном направлении. В этом примере я использую двунаправленный слой LSTM.

В примере обучается речевое обнаружение двунаправленной сетью LSTM с последовательными спектральными характеристиками и гармоническими метриками отношения.

Чтение звукового файла, который состоит из слов, произнесенных с паузами между. Зададим величину дискретизации аудио равной 16 кГц. Далее командой `audioread` идет чтение аудио сигнала. После командой `resample` идет передискретизация сигнала. И командой `sound(speech,fs)` идет воспроизведение аудио, после задается функция `win = hamming()` которая вызывает окно Хэмминга, которое дает вам более точное изображение частотного спектра исходного сигнала.

Листинг:

```

fs = 16e3;
[speech,fileFs] = audioread('Test1.wav');
speech = resample(speech,fs,fileFs);
speech = speech/max(abs(speech));
sound(speech,fs)
win = hamming(50e-3 * fs,'periodic');

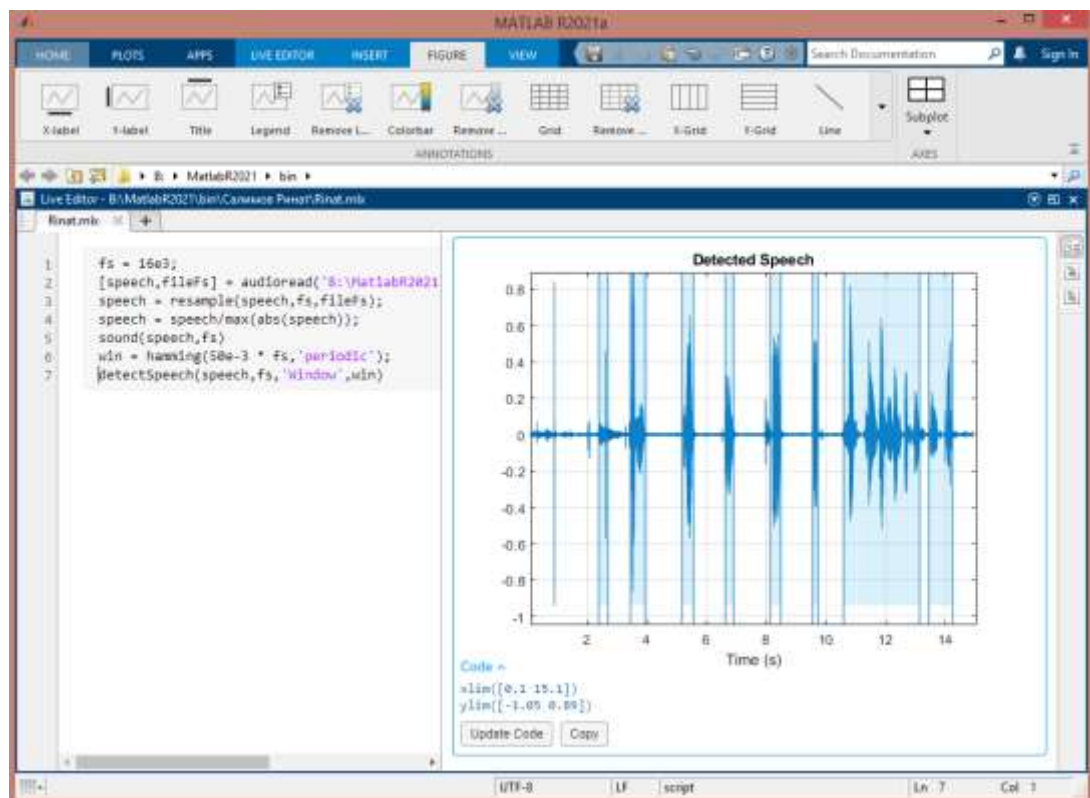
```

Далее используем функцию `detectSpeech` (Audio Toolbox), чтобы определить местоположение областей нет тишины и есть произношение слов. `DetectSpeech` - это функция которая правильно идентифицирует все области речи. Она использует алгоритм пороговой обработки сигнала на основе энергии и спектрального распространения на аналитическую систему координат. В функцию можно добавить алгоритм такой как `Window`, `OverlapLength`, и `MergeDistance` подстраивать их для ваших определенных потребностей. Результат сегментации приведен на рисунке 2.4.

```

detectSpeech(speech,fs,'Window',win)

```



Рисунке 2.4 – Сегментация речевого сигнала

Далее проверим сегментацию речи при зашумлении сигнала. Для этого необходимо добавить шум в аудио сигнал. Добавить шум в звуковой сигнал можно командой `resample(noise,fs,fileFs)` в размере ОСШ на-200 дБ. После вызовим функцию `detectSpeech` для зашумленного звукового сигнала.

Листинг:

```

[noise,fileFs] = audioread('TestNoise.wav');

```

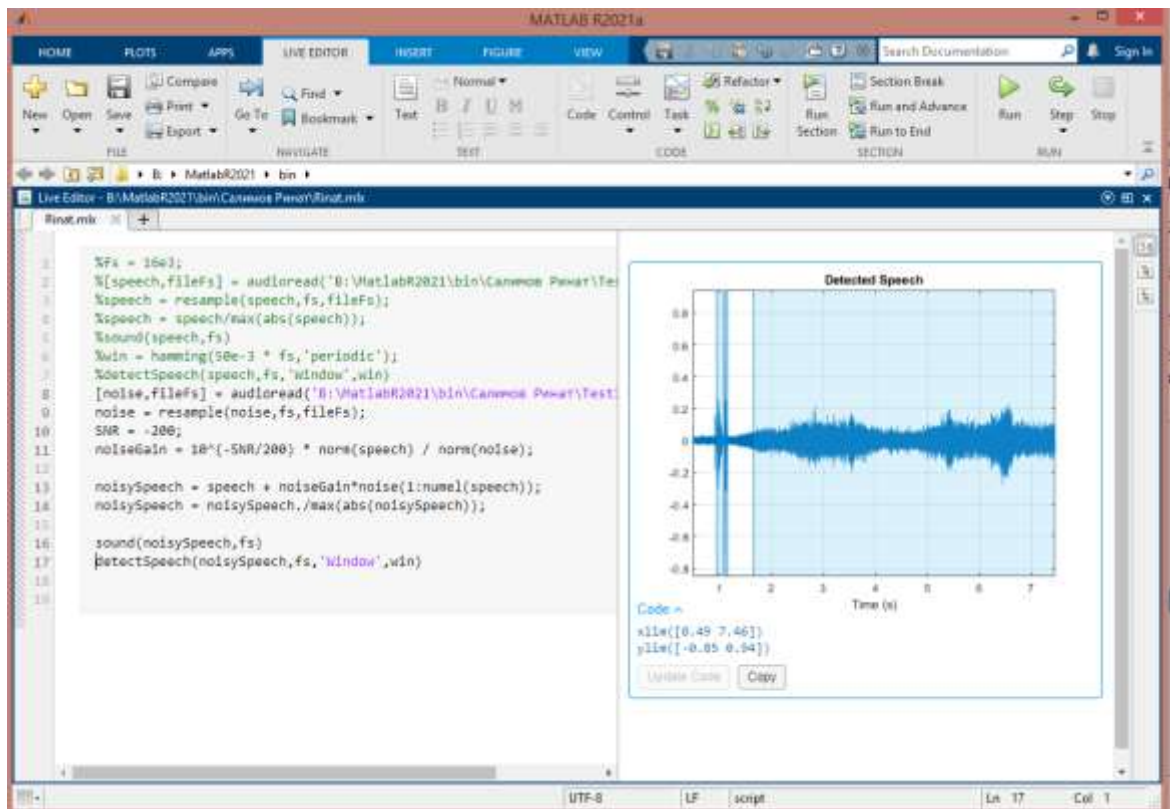


```

noise = resample(noise,fs,fileFs);
SNR = -200;
noiseGain = 10^(-SNR/200) * norm(speech) / norm(noise);
noisySpeech = speech + noiseGain*noise(1:numel(speech));
noisySpeech = noisySpeech./max(abs(noisySpeech));
sound(noisySpeech,fs)
detectSpeech(noisySpeech,fs,'Window',win)

```

Как видно на рисунке 2.5 функции не удастся обнаружить речевые области, учитывая очень низкий ОСШ.



Рисунке 2.5 – Сегментация зашумленного речевого сигнала

Для решения проблемы сегментации речи в зашумленном сигнале, была загружена предварительно обученная сеть. Она была обычная для обнаружения и сегментации речи в зашумленном сигнале, в среде с низким отношением сигнал шум. Эта функция входит в расширение audioFeatureExtractor для программы MatlabR2021.

Листинг:

```

url='http://ssd.mathworks.com/supportfiles/audio/VoiceActivityDetection.zip';
downloadNetFolder = tempdir;
netFolder = fullfile(downloadNetFolder,'VoiceActivityDetection');
if ~exist(netFolder,'dir')
    disp('Downloading pretrained network (1 file - 8 MB) ...')
    unzip(url,downloadNetFolder)

```

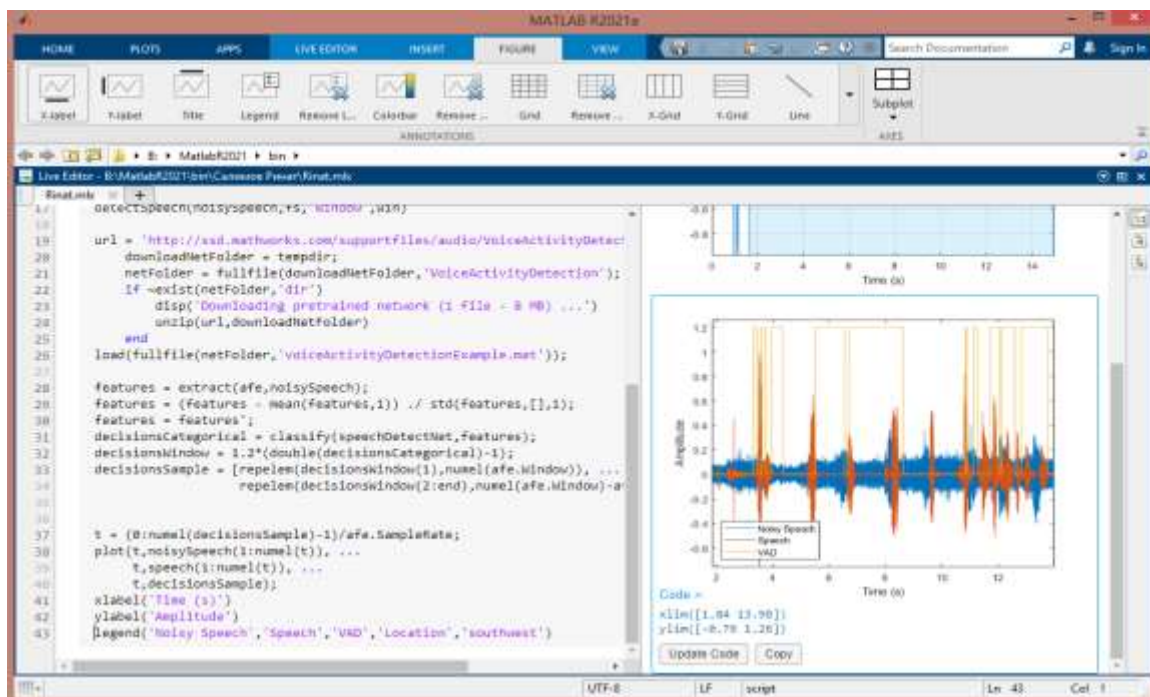


```

end
load(fullfile(netFolder,'voiceActivityDetectionExample.mat'));
Далее извлечем функции из речевых данных и затем нормируем их при
помощи команды extract(safe,noisy Speech). Сориентируем спектр так, чтобы
время распределялось по столбцам.
features = extract(afe,noisySpeech);
features = (features - mean(features,1)) ./ std(features,[],1);
features = features';
После функцию пропускаем через сеть по обнаружению речи, чтобы
классифицировать каждый вектор объектов как принадлежащий к фрейму
речи или же нет.
decisionsCategorical = classify(speechDetectNet,features);
Каждое решение соответствует аналитическому окну, анализируемому
audioFeatureExtractor. Реплицируем решения так, чтобы они были во взаимно-
однозначном соответствии с аудиосэмпами.
decisionsWindow = 1.2*(double(decisionsCategorical)-1);
decisionsSample = [repelem(decisionsWindow(1),numel(afe.Window)), ...
    repelem(decisionsWindow(2:end),numel(afe.Window)afe.OverlapLength)];
t = (0:numel(decisionsSample)-1)/afe.SampleRate;
plot(t,noisySpeech(1:numel(t)), ...
    t,speech(1:numel(t)), ...
    t,decisionsSample);
xlabel('Time (s)')
ylabel('Amplitude')
legend('Noisy Speech','Speech','VAD','Location','southwest')

```

После прохождения зашумленного звукового сигнала через специально обученную сеть был сегментирована речь. График сегментации речи с шумами показано на рисунке 2.6.



Рисунке 2.6 – Успешная сегментация зашумленного речевого сигнала

Далее разработаем Simulink-модель подсистемы сегментации сигналов, для получения спектров с фреймами, на рисунке 2.7. Эта модель использует блок Voice Activity Detector, чтобы визуализировать вероятность присутствия речи в звуковом сигнале. Использование механизма VAD (или подавления паузы) позволяет сэкономить на передаче данных по каналу связи, так как перерыв в речи (определяемый уровнем сигнала) не оцифровывается и не кодируется, и, таким образом, "пустые" пакеты с тишиной не передаются по сети. Таким образом, устранение "пустых" пакетов с небольшими шумами - это простой способ сэкономить трафик и, как следствие, увеличить пропускную способность канала.

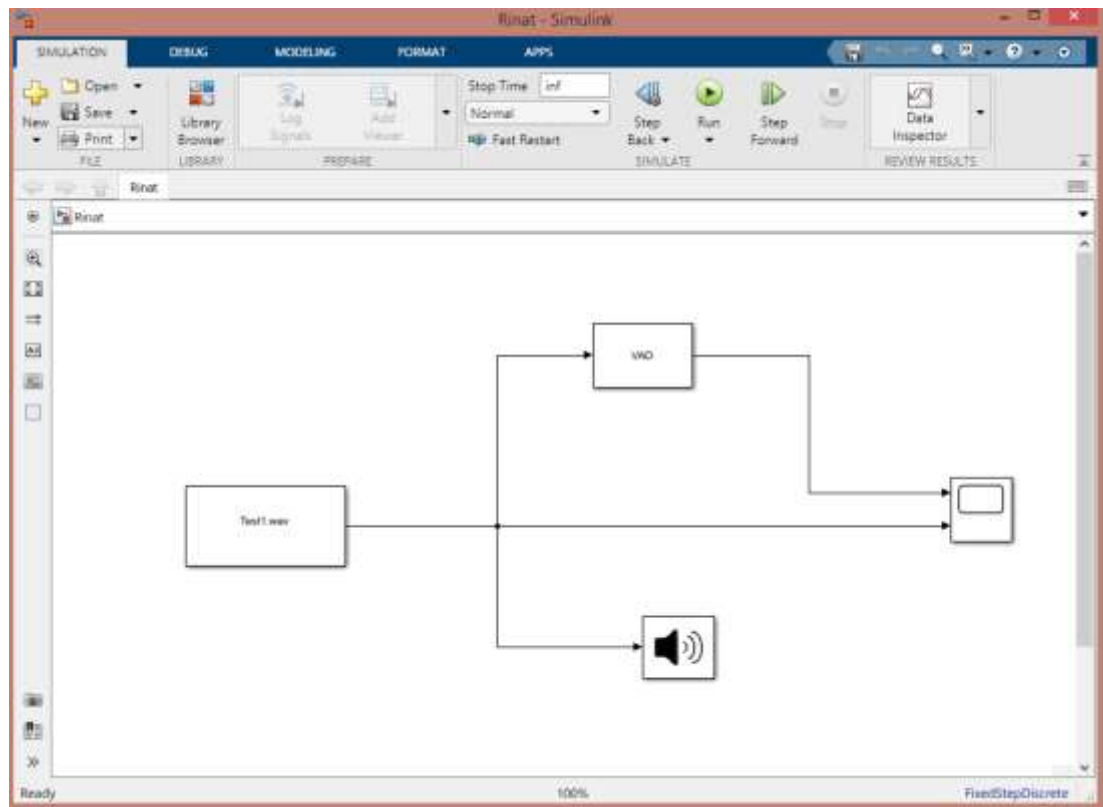


Рисунок 2.7 – Simulink-модель системы сегментации речи.

Эта модель чтобы визуализировать вероятность речевого присутствия в звуковом сигнале.

Далее введём в блок параметров аудио сигнал для сегментации в модели (рисунок 2.8). Если сигнал может иметь шумы то необходимо интегрировать в системе фильтры для борьбы с ними. После настроим речевой детектор действия (VAD) и наблюдаем вероятность присутствия слов и сегментацию речи который показан на рисунке 2.9.

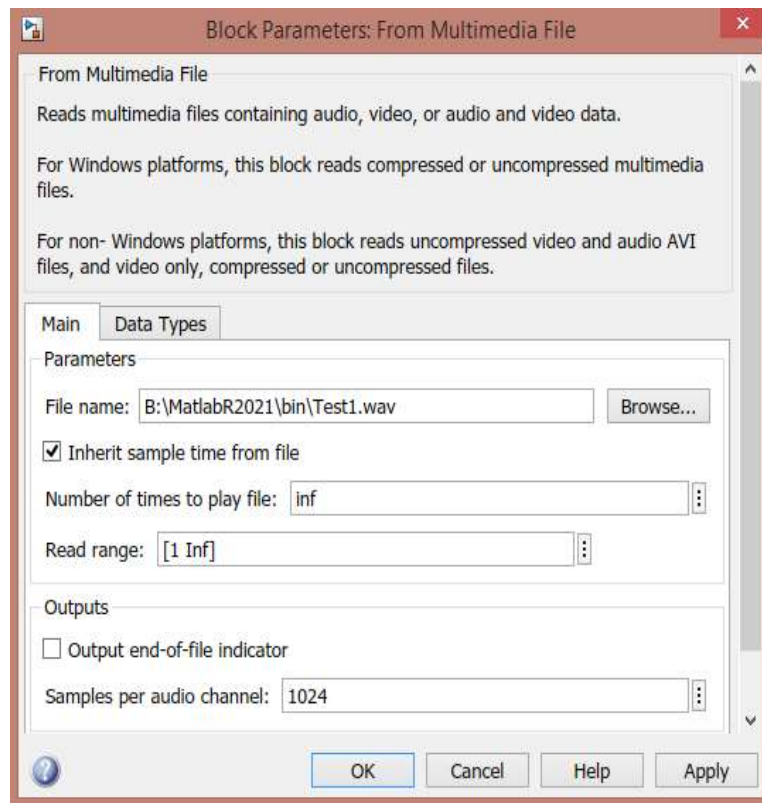


Рисунок 2.8 – Загрузка аудио сигнала в блок параметров.

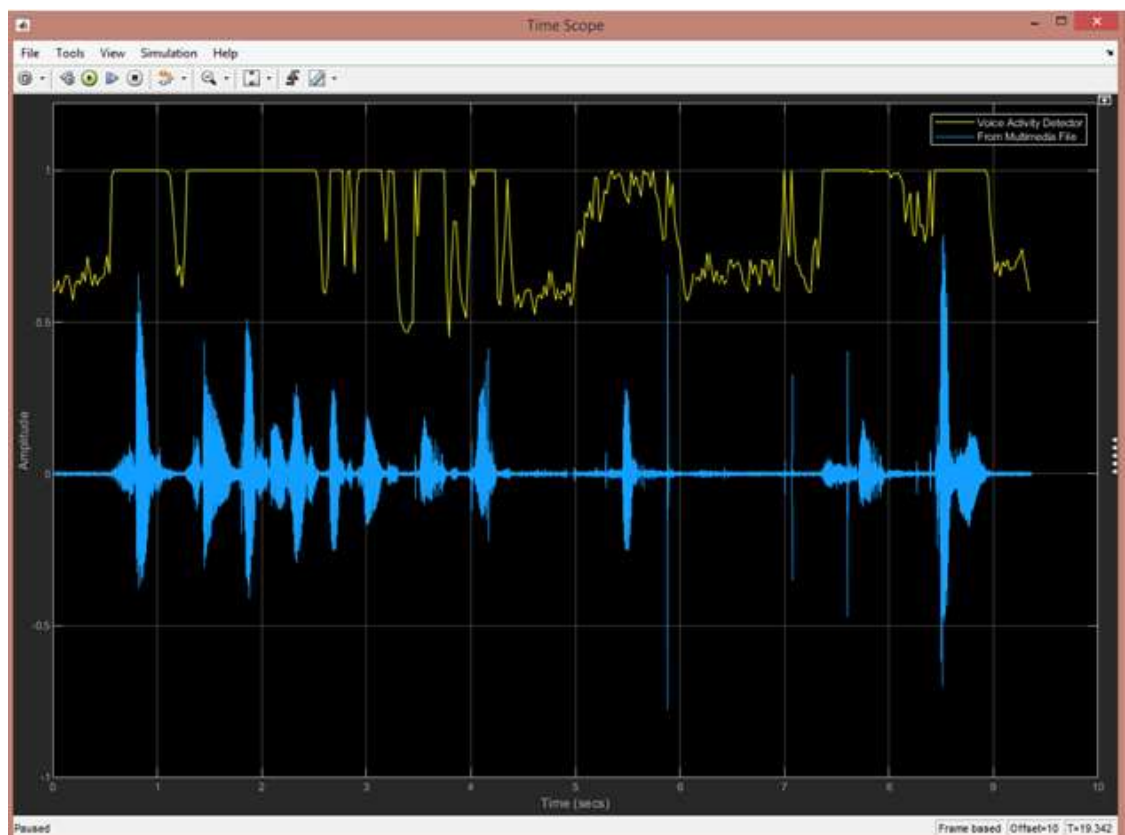


Рисунок 2.9 – Спектрограмма сегментации речи в Simulink .

2.12 Моделирование работы нейронной сети Кохонена в программе MATLAB

Нейронные сети Кохонена [9] нужны для решения задач векторной классификации. И до обучения будет не известно, сколько классов может использоваться для решения этой задачи, поэтому и требуется использовать обучение без учителя.

Сеть Кохонена – это искусственная нейронная сеть, которая содержит входной слой и слой активных нейронов. Из-за того что входной слой распределяет входные сигналы по нейронам активного слоя, такую сеть считают однослойной.

Активный слой может быть одномерным или двумерным. Это отражает представление в нейробиологии, с которыми сенсорная информация, представленная большим множеством сигналов, может отображаться в линейных или же планарных структурах коры головного мозга. Различают два вида сетей Кохонена:

1. Слой Кохонена. В нем нейроны активного слоя не упорядочены. В процессе обучения подстраиваются веса только одного нейрона-победителя.

2. Карта Кохонена (Self-Organized Map – SOM). В данном случае нейроны активного слоя образуют регулярную структуру. Такая карта применяется обычно для кластеризации графических изображений и звуковых сигналов.

Слой Кохонена в MATLAB. В процессе обучения SOM после предъявления достаточного количества образов все выходные нейроны в сети делятся на подмножества, каждое из которых может «откликаться» на образы соответствующего ему класса.

Архитектура соревновательного слоя приведена на рисунке 2.10. На рисунке 2.10, R – это размерность входного вектора p и векторов его весов, $S1$ – это количество нейронов слоя соревнования, а блок C (competitive layers) обозначает расчет выходов из нейронов с учетом их функции близости. Вектор смещений b изменяющийся в ходе обучения может использоваться для охвата нейронами всего входного пространства.

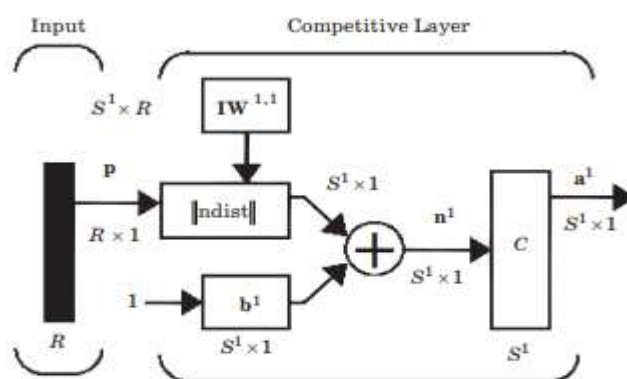


Рисунок 2.10. – Соревновательный слой Кохонена в MatLab

Слой Кохонена создается командой

```
net = newc(P, S, KLR, CLR),
```

где P – $(R \times Q)$ -матрица минимальных и максимальных значений Q входных векторов; S – число нейронов; KLR и CLR – константы скорости обучения весов и смещений.

Рассмотрим задачу кластеризации точек на плоскости. Пусть экспериментальные данные будут равны Разместим случайным образом 60 точек на плоскости с помощью команд:

```
A = [rand(1,20) - 0.7; rand(1,20) + 0.2];  
B = [rand(1,20) + 0.7; rand(1,20) + 0.7];  
C = [rand(1,20) + 0.2; rand(1,20) - 0.7];  
plot(A(1,:),A(2:,:), 'bs')  
hold on  
plot(B(1,:),B(2:,:), 'r+')  
plot(C(1,:),C(2:,:), 'go')  
grid on  
P = [A, B, C]  
ncl = 3;  
MN=[min(X(1,:)) max(X(1,:)); min(X(:,2)) max(X(:,2)) ]  
net = newc(MN, ncl, 0.1, 0.0005);  
net.trainParam.epochs=49;  
net.trainParam.show=7;  
net = train(net,P);  
w = net.IW{ 1 };  
plot(w(:,1),w(:,2), 'kp');
```

Результат обучения приведен на рисунке 2.11, где звездочками обозначены центры трех кластеров.

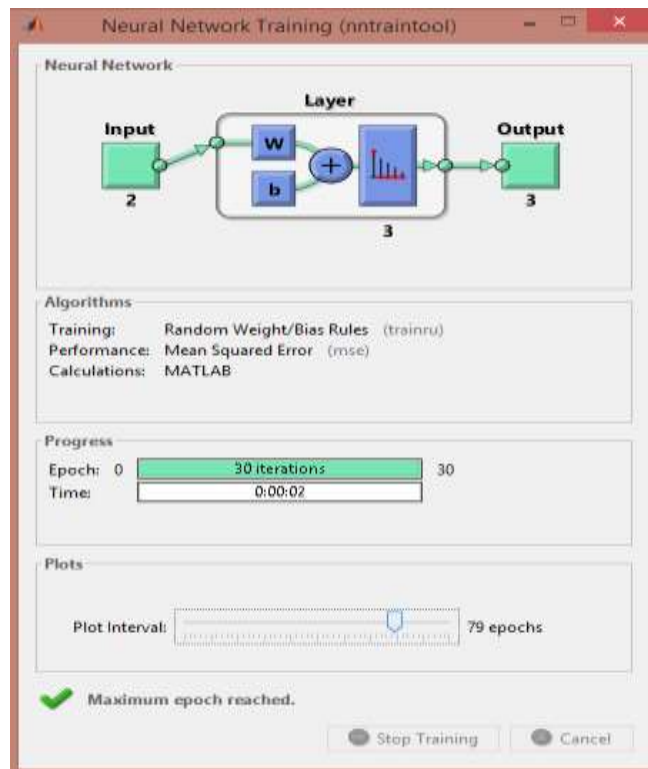


Рисунок 2.11 – Результат обучения нейросети с 3 нейронами

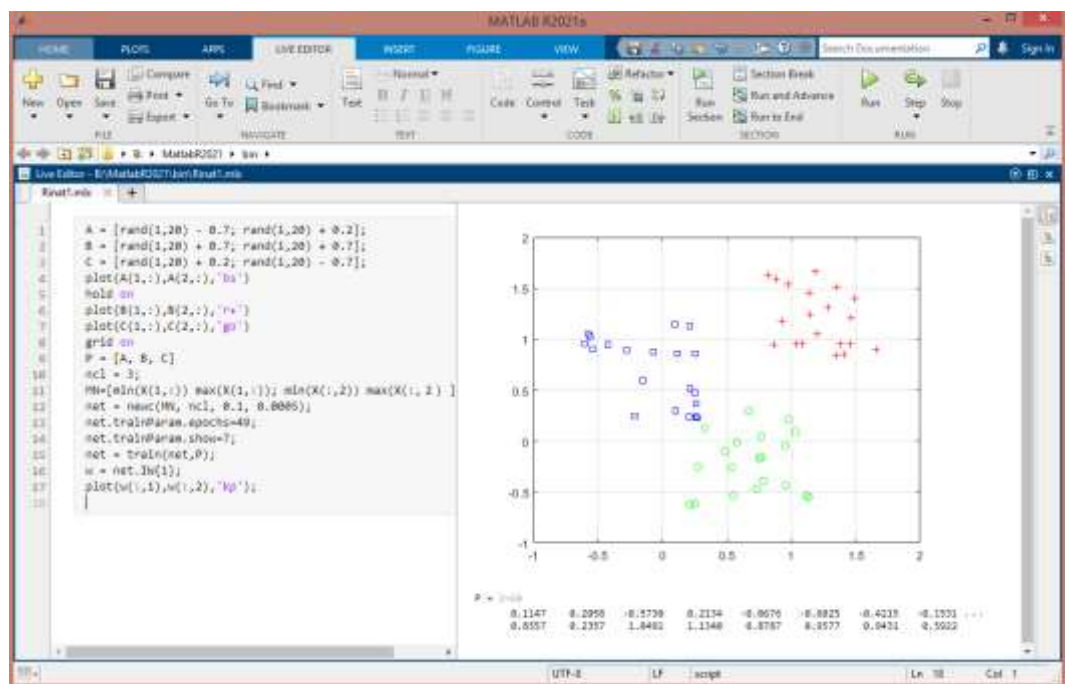


Рисунок 2.12 – Распределение трех кластеров на плоскости

Из рисунка 2.12 видно распределение результатов на три области так как нами было задано три нейрона. Для того чтобы увеличить плотность распределения результатов необходимо увеличить количество эпох обучения (рисунок 2.13). После мы получим распределение значений на плоскости результаты которых довольно плотно разбиты на три области (рисунок 2.14).

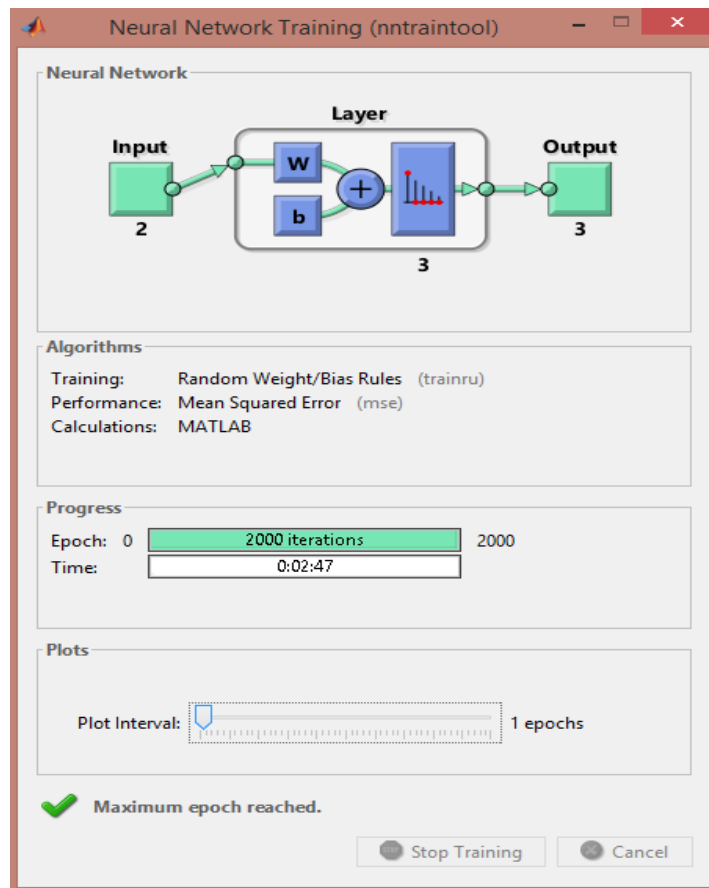


Рисунок 2.13 – Результат обучения нейросети с 3 нейронами

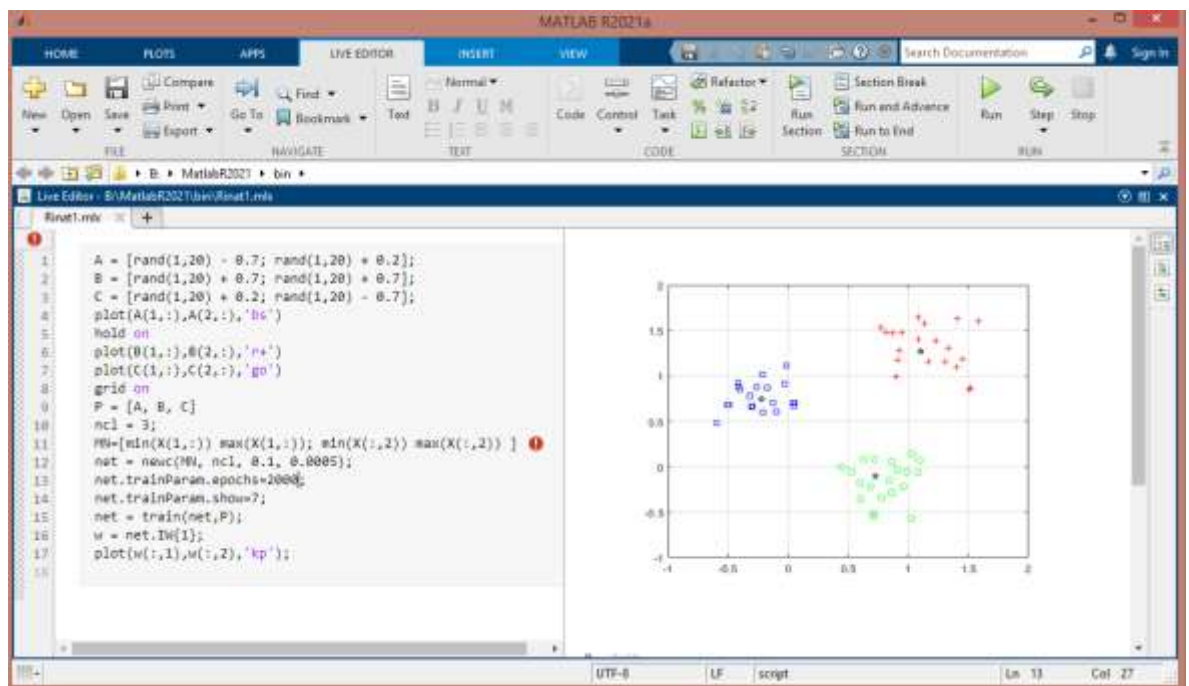


Рисунок 2.14 – Распределение трех кластеров на плоскости

Самоорганизующиеся карты Кохонена в MATLAB. Самоорганизующиеся карты Кохонена используются для визуализации многомерных данных. Конечно, невозможно без искажений спроецировать многомерный образец на плоскость, а SOM отражают только более широкую картину. Однако вы можете использовать SOM для анализа характеристик кластерной структуры многомерной выборки. Это может быть очень важно для качественного анализа информации..

Идея состоит в том, чтобы спроецировать все объекты образца на плоскую карту, точнее на набор узлов прямоугольной сетки заданного размера $M \times N$, который может быть порядка десятков или сотен. Чтобы карта отражала кластерную структуру образца, близлежащие объекты должны попасть в узлы плотной решетки. Нейроны SOM предназначены для равномерного распределения с равномерно расположенными входными векторами. Когда входные векторы распределены неравномерно, нейроны стремятся попасть в центр кластеров. Таким образом, SOM решает две проблемы:

- уменьшение размерности данных с минимальной потерей информации (анализ основных компонентов данных, выбор независимых характеристик);
- уменьшение разнообразия данных за счет выделения конечного набора прототипов и присвоения данных одному из этих типов (то есть кластеризации).

Самоорганизующаяся карта Кохонена создается с помощью команды `net = newsom(PR,[d1,d2,...], tfcn, dfcn, olr, osteps, tlr, tnd)`,

где PR – $(R \times 2)$ - матрица минимальных и максимальных значений R входных элементов; d_i – размерность сло; $tfcn$ – выбор топологии (по умолчанию 'hextop'); $dfcn$ – функция дистанции (по умолчанию 'linkdist'); olr – параметр скорости обучения на этапе размещения; $osteps$ – число циклов обучения на этапе подстройки ; tlr – параметр скорости на этапе подстройки; tnd – размер окрестности на этапе подстройки.

Функция `hextop` соответствует гексагональной, `gridtop` – прямоугольной и `randtop` – случайной топологии.

Функция дистанции `dist` соответствует евклидову расстоянию, `boxdist` – максимальное координатное смещение, `mandist` – расстояние суммарного координатного смещения, `linkdist` – расстояние связи.

Настройка SOM производится по каждому входному вектору. Прежде всего определяется нейрон-победитель и корректируются его вектор весов и векторы соседних нейронов согласно соотношению:

$$dW = lr \cdot A2 \cdot (P' - W) \quad (2.6)$$

где lr – параметр скорости обучения, равный olr для этапа упорядочения нейронов и tlr – для этапа подстройки;

$A2$ – массивы соседства для нейронов, расположенных в окрестности нейрона-победителя с номером i :

$$A2(i, q) = \begin{cases} 1 & a(i, q) = 1, \\ 0,5 & a(i, q) = 1, D(i, j) \leq nd \\ 0 & \text{в остальных случаях} \end{cases} \quad (2.7)$$

Здесь $a(i, q)$ – элемент выхода нейронной сети; $D(i, j)$ – расстояние между нейронами i и j ; nd – размер окрестности нейрона-победителя.

Таким образом, весовое изображение нейрона-победителя изменяется пропорционально половине параметра скорости обучения, а весовое значение соседних нейронов изменяется пропорционально половине значения этого параметра.

Весь процесс обучения карты Кохонена делится на этапы:

- упорядочения векторов весовых коэффициентов в пространстве признаков;
- подстройки весов нейронов по отношению к набору векторов входа.

На этапе упорядочения используется фиксированное количество шагов. Начальный размер окрестности устанавливается равным максимальному расстоянию между нейронами для выбранной топологии, а затем уменьшается до значения, используемого на следующем шаге. Рассчитывается по формуле 2.8:

$$nd = 1,000001 + (\max(d) - 1)(1 - \frac{s}{S}) \quad (2.8)$$

где $\max(d)$ – максимальное расстояние между нейронами; s – номер текущего шага; S – число циклов на этапе упорядочения. Параметр скорости обучения изменяется по формуле 2.9.

$$lr = tlr + (olr - tlr)(1 - \frac{s}{S}) \quad (2.9)$$

На этапе подстройки, который продолжается в течение оставшейся части процедуры обучения, размер окрестности остается постоянным и равным

$$nd = tnd + 0,00001 \quad (2.10)$$

Нейроны карты Кохонена упорядочены таким образом, что при одинаковой плотности входных векторов эти нейроны также распределяются равномерно. Если входные векторы распределены неравномерно, нейроны также имеют тенденцию распределяться в соответствии с плотностью входных векторов.

Кластеризация случайно заданного массива из 60 точек на плоскости:

$P = [\text{rand}(1,60)*2; \text{rand}(1,60)];$

```

net = newsom([0 2; 0 1],[3 5]);
net = train(net,P);
plot(P(1,:),P(2:,:),'.g','markersize',20)
hold on
plotsom(net.iw{1,1},net.layers{1}.distances)
plotsompos(net,P);
plotsomnd(net);
plotsomhits(net2,P);

```

Результат кластеризации приведен на рисунке 2.15. В MatLab имеются дополнительные средства анализа SOM. Функция `plotsomnd(net)` порождает график, аналогичный показанному на рисунке 2.15. Функция позволяет визуализировать дистанцию между весами нейронов (рисунке 2.16).

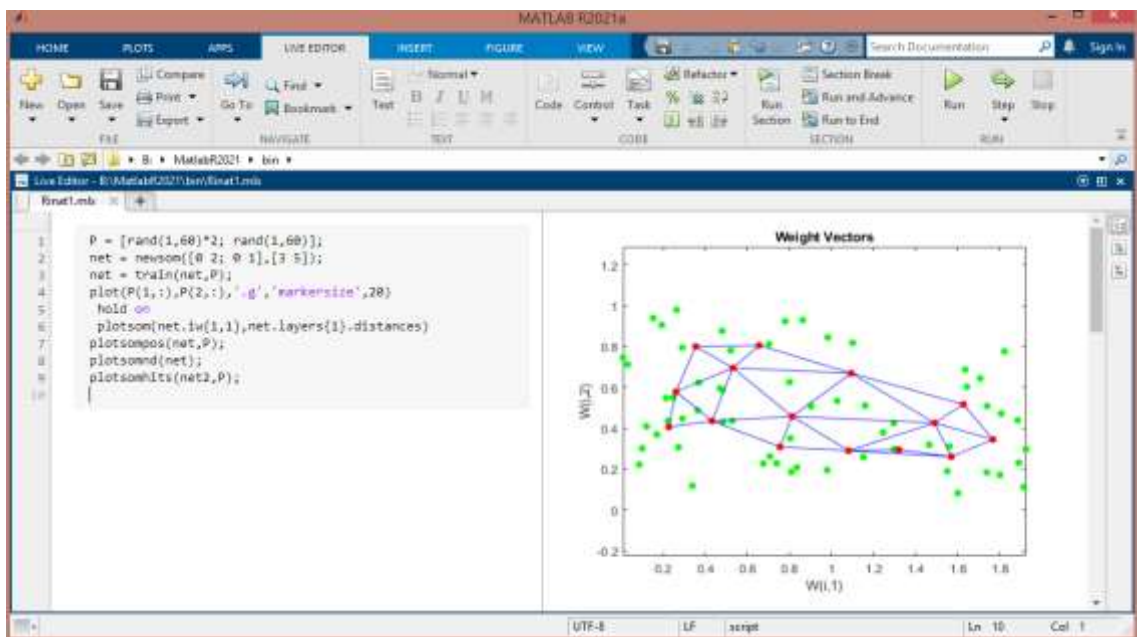


Рисунок 2.15 – Использование двумерной SOM

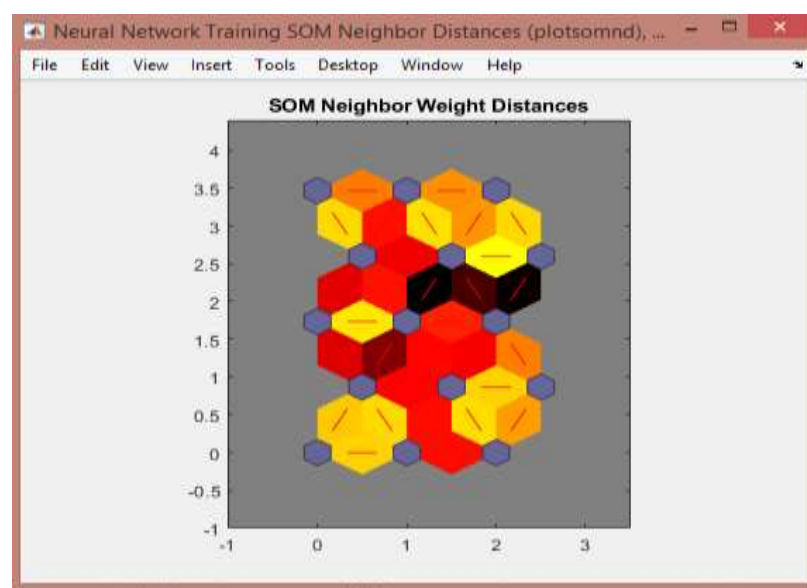


Рисунок 2.16 – Визуализация дистанции между весами SOM

Еще одна функция позволяет узнать число входных векторов, на которые реагирует каждый нейрон: `plotsomhits(net2,P)` Пример приведен на рисунке 2.17

Удобным инструментом визуализации данных является раскраска топографических карт аналогично тому, как это делают на обычных географических картах. Каждый признак данных порождает свою раскраску ячеек карты – по среднему значению этого признака у данных, попавших в данную ячейку.

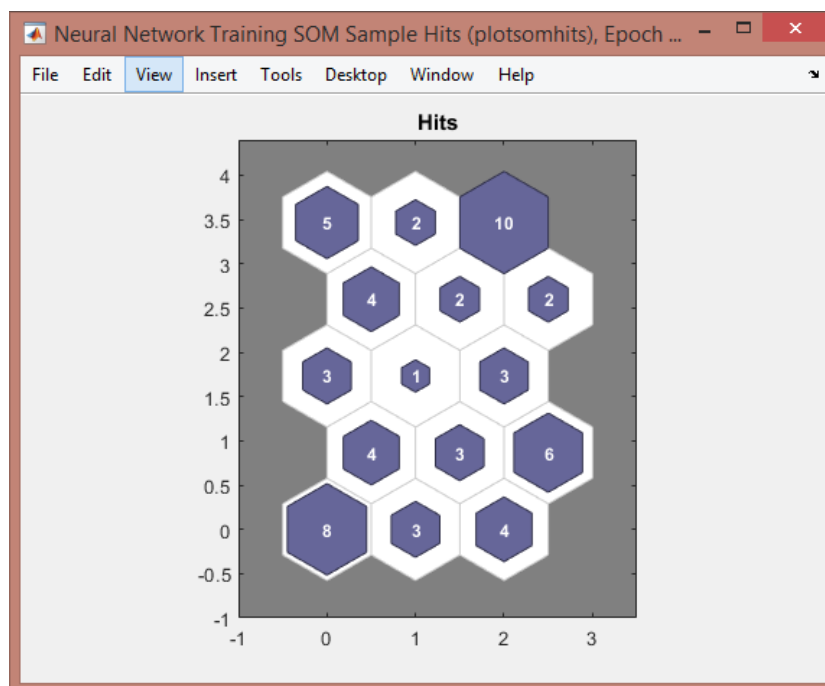


Рисунок 2.17 – Визуализация числа векторов, ассоциированных с нейронами двумерной SOM

2.13 Реализация распознавания речи

Распознавание речи будет реализовано с помощью Python. В ходе реализации проекта в качестве базы данных с записанными речами. Так же одним из плюсов является работа данной в режиме офлайн. На рисунке 2.18 показан листинг кода в среде Python.

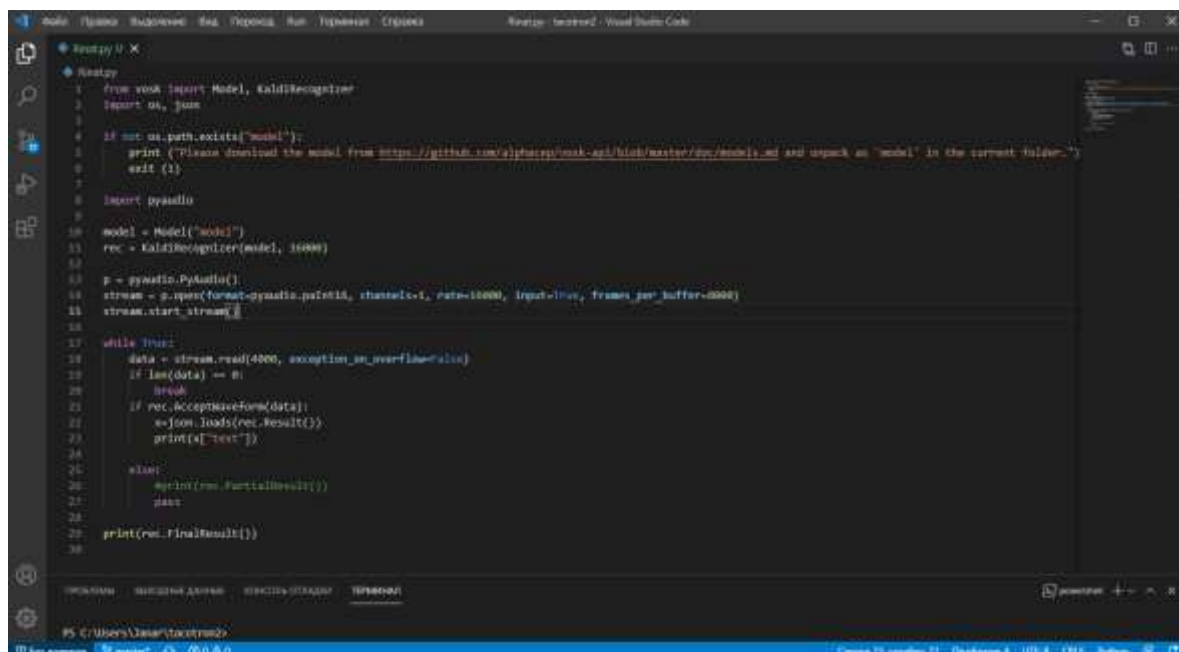


Рисунок 2.18 – Листинг кода Python для распознавания речи

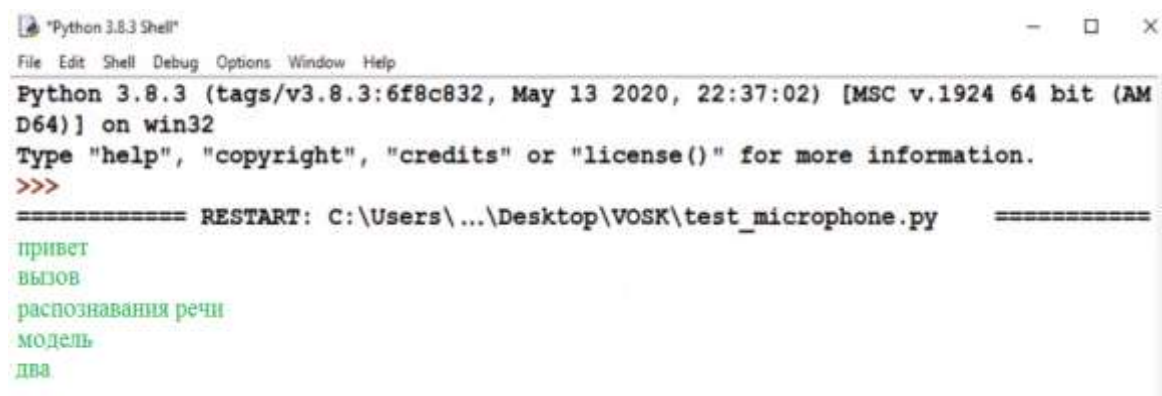


Рисунок 2.19 – Пример работы программы для распознавания речи

3 Безопасность жизнедеятельности

3.8 Анализ условий труда вычислительного центра

В этом разделе дипломного проекта приводится анализ условий труда в вычислительном центре. Тема дипломной работы: «Реализация технологии распознавания речи с использованием передовых архитектур ИНС (искусственная нейронная сеть)».

Тип помещения – зал вычислительного центра (ВЦ) который находится в центре первого этажа. Длинной 10 м., шириной в 5 м. и высотой стен в 3 м.. Количество рабочих мест – 4. В зале размещено серверное оборудование. По стандартам и рекомендациям в серверном кабинете нет окон. Нормативные показатели рекомендуемой температуры воздуха 20-25 градусов. В серверной будет система поддержки климата, она должна работать постоянно (кондиционер или другое оборудование для охлаждения). В зале ВЦ размещено оборудование: электронно-вычислительные машины 6 шт.

3.9 Расчет системы кондиционирования воздуха

Необходимо рассчитать избыточное тепло. В помещение общее избыточное тепло поступает от следующих источников:

- $Q_{изб1}$ – тепло от электрооборудования и освещения;
- $Q_{изб3}$ – тепло, излучаемое людьми;
- $Q_{изб4}$ – тепло, излучаемое оборудованием.

Тепло, излучаемое от освещения, рассчитывается следующим образом:

Коэффициент перехода электрической энергии в тепловую энергию для люминесцентных ламп: $\eta = 0,5 - 0,6$

Установленная мощность использованы накладных светильников:

$$N_{осв} = 36 \text{ Вт/м}^2.$$

Площадь пола

$$S_{пол} = 10 \cdot 5 = 50 \text{ м}^2$$

Тогда

$$Q_{осв} = 0,5 \cdot 36 \cdot 50 = 900 \text{ Вт.}$$

Количество тепла, выделяемое человеком, рассчитывается следующим образом [11]:

$$Q_{изб3} = n \cdot q \quad (3.1)$$

где n – число людей в офисе, $n = 4$ работников на одну смену;

$q = 150 \text{ Вт/чел}$ – тепло выделяемое одним человеком в соответствии с категорией сложности работы (Приложение А, табл. 2)

$$Q_{\text{изб3}} = 4 \cdot 150 = 600 \text{ Вт.}$$

Количество тепла, выделяемого оборудованием [16]:

$$Q_{\text{изб4}} = N \cdot n_1 \cdot n_2 \cdot n_3 \cdot n_4 \quad (3.2)$$

Тепло, выделяемое производственным оборудованием, определяется по формуле (4.3) [1]:

$$Q_{\text{об}} = N_{\text{уст}} \cdot K \quad (3.3)$$

$$Q_{\text{об}} = 2 \cdot 6 \cdot 0,9 = 10,8 \text{ кВт.}$$

Количество тепла, поступающего в помещение в результате разности температур определяется по формуле:

$$Q_{\text{огр}} = V_{\text{пом}} \cdot X_0 \cdot (t_{\text{Нрасч}} - t_{\text{Врасч}}) \quad (3.4)$$

где $V_{\text{пом}} = 10 \cdot 5 \cdot 3 = 150 \text{ м}^3$ – объём помещения;

$X_0 = 0,42 \text{ Вт/м}^3 \text{ } ^\circ\text{С}$ – удельная тепловая характеристика;

$t_{\text{Нрасч}} = 27^\circ$ - расчётная наружная температура для тёплого периода года, табл;

$t_{\text{Нрасч}} = -10^\circ$ - расчётная наружная температура для холодного периода года;

$t_{\text{Врасч}} = 24^\circ$ - расчётная внутренняя температура для тёплого периода года;

$t_{\text{Врасч}} = 20^\circ$ - расчётная внутренняя температура для холодного периода года.

Тогда теплоступление для тёплого периода года составят:

$$Q_{\text{огрЛетом}} = 150 \cdot 0,42 \cdot (27 - 24) = 189 \text{ Вт}$$

А теплототери для холодного периода года составят:

$$Q_{\text{огрЛетом}} = 150 \cdot 0,42 \cdot |-10 - 20| = 1890 \text{ Вт}$$

В холодное время года, также надо рассчитать теплоступление от отопления. Необходимо иметь возможность отключать отопление в серверной зимой и осенью (летом оно не работает по умолчанию). Для упрощения по считаем теплотприток от системы центрального отопления 100 Вт/м^2 или $0,1 \text{ кВт/м}^2$. Итого теплотприток от системы центрального отопления [16]:

$$Q_{\text{отоп}} = S \cdot 100 = 50 \cdot 100 = 5000 \text{ Вт}$$

Таким образом, для определения потребного воздухообмена необходимо определить количество избыточного тепла по формуле:

$$Q_{\text{изб}} = Q_{\text{об}} + Q_{\text{осв}} + Q_{\text{л}} + Q_{\text{отоп}} + Q_{\text{отд}}, \text{ ккал/ч} \quad (3.5)$$

где $Q_{\text{об}}$, ккал/ч – тепло, выделяемое оборудованием;

$Q_{\text{осв}}$, ккал/ч – тепло, выделяемое системой освещения;

$Q_{\text{л}}$, ккал/ч – тепло, выделяемое людьми в помещении;

$Q_{\text{отоп}}$, ккал/ч – тепло, вносимое за счет отопления (в холодный период);

На основании выполненных расчетов составим баланс теплоступлений в помещении.

Для тёплого периода года:

$$Q_{\text{изб}} = 10800 + 900 + 600 + 189 = 12489 \text{ Вт}$$

Для холодного периода года:

$$Q_{\text{изб}} = 10800 + 900 + 600 + 1890 + 5000 = 19190 \text{ Вт}$$

Так как тепловой баланс для лета больше зимнего теплового баланса, то рассчитаем теплонапряженность воздуха по формуле [11]:

$$Q_{\text{н}} = \frac{Q_{\text{изб.лето}} \cdot 860}{V_{\text{пом}}} = \frac{19190 \cdot 860}{150} = 110 \text{ ккал/м}^3. \quad (3.6)$$

Рассчитаем количество избыточного тепла, которое необходимо вывести из помещения [15]:

$$L_{\text{б}} = Q_{\text{изб}} / (C_{\text{в}} \cdot \Delta t \cdot \rho_{\text{возд}}) \quad (3.7)$$

где $Q_{\text{изб}}$ – рассчитанное выше избыточное тепло, ккал/ч;

$C_{\text{в}} = 0,24 \text{ Вт/кг} \cdot ^\circ\text{C}$ – теплоемкость воздуха;

Δt – разница между температурами выходящего и поступающего воздуха, $^\circ\text{C}$;

$\gamma = 1,206 \text{ кг/м}^3$ – удельная масса приточного воздуха

Δt определяется исходя из значения тепло напряженности воздуха [15]:

$$\Delta t = t_{\text{вых}} - t_{\text{пр}}, ^\circ\text{C} \quad (3.8)$$

где $t_{\text{вых}}$, $^\circ\text{C}$ – температура уходящего воздуха;

$t_{\text{пр}}$, $^\circ\text{C}$ – температура приточного воздуха.

Величина Δt при расчетах выбирается в зависимости от теплонапряженности воздуха – $Q_{\text{н}}$:

при $Q_{\text{н}} \leq 20 \text{ ккал/м}^3 \cdot \text{ч}$ $\Delta t = 6 ^\circ\text{C}$;

при $Q_{\text{н}} > 20 \text{ ккал/м}^3 \cdot \text{ч}$ $\Delta t = 8 ^\circ\text{C}$;

Из $Q_{\text{н}} = 16 \text{ ккал/м}^3$ следует, что оно меньше $20 \text{ ккал/м}^3 \cdot \text{ч}$, а значит, что разница между температурами выходящего и поступающего воздуха $\Delta t = 6 ^\circ\text{C}$ [15].

Подставляем все данные в формулу

$$L = \frac{Q_{\text{изб}} \cdot 860}{C \cdot \Delta t \cdot \gamma} = \frac{19190 \cdot 860}{0.24 \cdot 8 \cdot 1.206} = 7127 \text{ м}^3/\text{час} \quad (3.9)$$

Определение кратности воздухообмена[11]:

$$n = \frac{L}{V_{\text{пом}}} = \frac{7127}{150} = 48 \text{ час}^{-1}. \quad (3.10)$$

Исходя из расчетов, в рабочем помещении необходима установка кондиционера. Для установки выбираем кондиционер LG DM 09 RP. Данные кондиционеры рассчитаны на покрытие площади от 50 до 55 м², объемом от 150 м³. Мощность в режиме охлаждения равна 4815 Вт, а потребляемая мощность при охлаждении 1495 Вт. При обогреве мощность будет соответственно больше – 5315 Вт, мощность обогрева – 1468 Вт. Данный кондиционер создает потокообмен в 650-1000 м³/ч. И возможно нам необходимо несколько таких.



Рисунок 4.1 – Кондиционер LG DM 09 RP

Таблица 3.1- Технические характеристики кондиционера LG DM 09 RP

Площадь помещения, кв.м.:	50
Тип кондиционера:	настенная сплит-система
Температура в режиме нагрева, С:	16/30
Температура в режиме охлаждения, С:	18/30

Мощность в режиме охлаждения/обогрева, Вт:	2800/3500
Потребляемая мощность охлаждения/обогрев, Вт:	556/712
Уровень шума внутреннего блока, дБ:	19
Класс энергопотребления:	A++
Циркуляция:	поток воздуха в 4-х направлениях
Тип компрессора:	инверторный
Тип фильтра:	дезинфицирующий
Бренд:	LG
Страна производитель:	Южная Корея

На рисунке 3.2 показано схема расположения кондиционера в производственном помещении

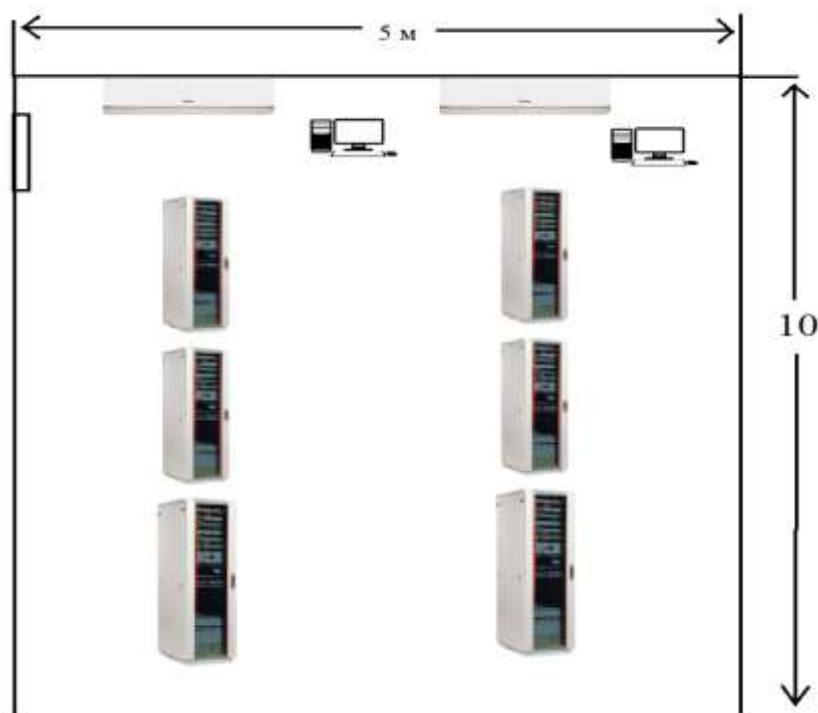


Рисунок 3.2 – Схема расположения кондиционера в производственном помещении

3.10 Разработка эвакуационных путей

Для обеспечения безопасности эвакуации в проекте рассчитано время эвакуации людей из рабочих помещений. В главном офисе банка работают – 618 человек.

Здание 1 этажное, с размерами в коридорах шириной 3 м, в которых есть схема эвакуации людей при пожаре на каждом этаже, как показано на рисунке

3.2. Серверная объемом 150 м³ расположен на первом этаже в прямой близости к пожарному выходу.

Плотность людского потока D , состоящая из N людей, равна [16]:

$$D = N/A; \quad (3.11)$$

где A — площадь пути эвакуационного участка, м²;

$$D = \frac{35}{375} = 0.09$$

$$A = \delta l;$$

$$A = 15 * 20 = 375$$

Пропускной способностью пути называют количество людей, проходящих в единицу времени через поперечное сечение пути шириной δ [16]:

$$Q = Dv \delta; \quad (3.12)$$

$$Q = 0.08 * 100 * 3.7 = 12.7$$

Интенсивностью движения людского потока называется величина, равная произведению плотности и скорости движения [16]:

$$q = Dv; \quad (3.13)$$

$$q = 0.08 * 100 = 8$$

Интенсивность движения не зависит от ширины пути и является характеристикой потока. Поскольку скорость движения потока есть функция его плотности, то интенсивность движения также является функцией плотности.

Расчетное время эвакуации людей t_p определяется как сумма времени движения людского потока по отдельным участкам пути [12]:

$$t_p = t_{н.э} + t_1 + t_d + t_2 + t_3 \dots t_i; \quad (3.14)$$

где t_1 — время движения людского потока на начальном участке, мин;

$t_2, t_3, \dots t_i$ — период перемещения потока в любом из последующих после 1-ого участка пути, мин.

Время движения людского потока по первому участку пути [12]:

$$t_1 = L_1/V_1; \quad (3.15)$$

где V_1 — скорость движения людского потока по горизонтальному пути на первом участке определяется по таблице 3.3 в зависимости от плотности D_1 , м/мин.

Плотность людского потока D_l на первом участке пути, имеющем длину l_l , и ширину δ_l , равна [16]:

$$D=N_l \cdot f / l_l \cdot b_l ; \quad (3.16)$$

где N_l — количество людей на первом участке;
 f — средняя площадь горизонтальной проекции человека, принятая по таблице 3.2 [16]:

Таблица 3.2 – Площади горизонтальной проекции человека

Возраст, одежда человека и вид груза		Площадь горизонтальной проекции человека f , м ²
Взрослый человек:	В летней одежде	0,1
	В демисезонной одежде	0,113
	В зимней одежде	0,125
	С ребенком на руках	0,285
	С рюкзаком	0,315
Подросток		0,07
Ребенок		0,04-0,05

Таблица 3.3 –Скорость движения людского потока

Плотность потока D м ² /м ²	Горизонтальный путь		Дверной проем, интенсивность q , м/мин	Лестница вниз		Лестница вверх	
	Скорость V , в/мин	Интенсивность q , м/мин		Скорость V , м/мин	Интенсивность q , м/мин	Скорость V , м/мин	Интенсивность q , м/мин
0,01	100	1,0	1,0	100	1,0	60	0,6
0,05	100	5,0	5,0	100	5,0	60	3,0
0,10	80	8,0	8,7	95	9,5	53	5,3
0,20	60	12,0	13,4	68	13,6	40	8,0
0,30	47	14,1	16,5	52	15,6	32	9,6
0,40	40	16,0	18,4	40	16,0	26	10,4
0,50	33	16,5	19,6	31	15,6	22	11,0
0.6-0,9 и более	15	13,5	8,5	8	7,2	11	9,9

С целью расчета времени перемещения людей по начальному участку, учитывая габаритные размеры кабинета 6х6 м, обуславливается плотность перемещения людского потока на начальном участке по формуле (3.11):

$$\begin{aligned}D_1 &= 4 * 0.1 / 5 * 5 = 0.016 \text{ м}^2/\text{м}^2 \\D_2 &= 10 * 0.1 / 10 * 5 = 0.02 \text{ м}^2/\text{м}^2 \\D_3 &= 4 * 0.1 / 10 * 5 = 0.008 \text{ м}^2/\text{м}^2 \\D_4 &= 5 * 0.1 / 6 * 5 = 0.017 \text{ м}^2/\text{м}^2 \\D_5 &= 5 * 0.1 / 7 * 5 = 0.014 \text{ м}^2/\text{м}^2 \\D_6 &= 5 * 0.1 / 8 * 5 = 0.012 \text{ м}^2/\text{м}^2 \\D_7 &= 35 * 0.1 / 25 * 5 = 0.028 \text{ м}^2/\text{м}^2\end{aligned}$$

Тогда расчетное время эвакуации для каждого участка пути:

$$\begin{aligned}t_1 &= 5 / 100 = 0,05 \text{ мин.} \\t_2 &= 10 / 100 = 0,1 \text{ мин.} \\t_3 &= 10 / 100 = 0,1 \text{ мин.} \\t_4 &= 6 / 100 = 0,06 \text{ мин.} \\t_5 &= 7 / 100 = 0,07 \text{ мин.} \\t_6 &= 8 / 100 = 0,08 \text{ мин.} \\t_7 &= 25 / 100 = 0,25 \text{ мин.}\end{aligned}$$

Тогда расчетное время эвакуации, мин:

$$T_p = 0,05 + 0,1 + 0,1 + 0,06 + 0,07 + 0,08 + 0,25 = 0,71 \text{ мин.}$$

3.11 Расчет количества и типа огнетушителей

В помещении необходим дымовой пожарный извещатель (ДИП). При высоте помещения в 3 м, один извещатель контролирует площадь в 10 м². Количество ДИП определяется по формуле:

$$N_{\text{дип}} = S_{\text{пом}}/S_{\text{дип}} \quad (3.17)$$

$$N_{\text{дип}} = 50/10 = 5 \text{ извещателя}$$

Используем ДИП ИП 212-45 и пульт извещения «Топаз – 3М».

Количество огнетушителей зависит:

- 1) от предельно защищаемой площади одним огнетушителем или группой огнетушителей;
- 2) от максимального расстояния от возможного очага пожара до ближайшего огнетушителя, для помещений не должно превышать 20 м;
- 3) от размещения технологического оборудования в помещениях. Громоздкое технологическое оборудование уменьшает предельно допустимое расстояние от огнетушителя до возможного очага пожара;
- 4) от расположения стен и перегородок, а также геометрии помещения;

5) от экономических соображений. Например, в помещениях с большой площадью необходимо устанавливать большей ёмкостью огнетушители.

Участок приёмки и комплектации грузов с высотой размещения горючей нагрузки 3,0 м имеет категорию пожарной опасности В-1, класс пожара «А» площадь 50 м² ;

А) Определим примерное количество огнетушителей в зависимости от предельной площади покрытия одним К-огнетушителем[18]:

$$N_{оп1} = S_{п} : S_{о} \quad (3.18)$$

где:

$N_{оп1}$ – примерное количество огнетушителей

K – количество огнетушителей на одну зону – 2 ОП-8,

$S_{п}$ – площадь размещения горючей нагрузки помещения

$S_{о}$ – предельно защищаемая площадь одним огнетушителем.

$S_{о} = 18.6 \text{ м}^2$

$$N_{оп1} = 50 / 20 = 2.5$$

Требуется 3К ручных огнетушителей. Принимаем количество огнетушителей: 3 шт. ОП-8. В таблице 3.2 приведены технические характеристики порошкового огнетушителя. В моем проекте использовались порошковые огнетушители.

Таблица 3.4 –Технические характеристики порошкового огнетушителя ОП-8 [13]:

Марка огнетушителя		порошковый ОП-8
Огнетушащая способность (площадь, м2)		4А(18,66);144В(4,52)
Длина выброса, м		4,5
Масса, кг		11...12
Габариты, мм		585x260x230
Классы тушимых пожаров		А В С Е

На рисунке 3.3 показаны огнетушитель ОП - 8



Рисунок 3.3 – Огнетушитель ОП - 8

Углекислотный огнетушитель – это закачной огнетушитель высокого давления с зарядом жидкой двуокиси углерода, находящийся под давлением ее насыщенных паров. Углекислотный огнетушитель — один из видов первичных средств пожаротушения. Его баллон заполнен составом двуокиси углерода, находящегося под высоким давлением закаченного внутрь газа. Применение углекислотных огнетушителей широко распространено в промышленности и быту.

Огнетушители как первичные средства пожаротушения, заполненные углекислотой, незаменимы как средство тушить пожары там, где с другим видом огнетушащего вещества это сделать невозможно, смертельно опасно для жизни или нецелесообразно использовать из-за попадания на дорогостоящее и ценное производственное оборудование, электрическую аппаратуру, приборы, бытовую технику воды, химической пены, порошка, что приводило еще к большему материальному ущербу. Напротив, CO_2 в ходе тушения просто быстро испаряется, не оставляя абсолютно никаких следов – загрязнений и повреждений.

Таким образом, из полученных результатов можно сделать вывод, что для обеспечения нормального функционирования системы автоматического пожаротушения потребуется 8 баллонов углекислотно- хладонового состава вместимостью 20 литров, с массой смеси 6 кг для одного рабочего помещения. Автоматические установки газового пожаротушения имеют устройства для автоматического пуска в соответствии с ГОСТ 12,4.009-83. Также произведен расчет и план эвакуационных путей.

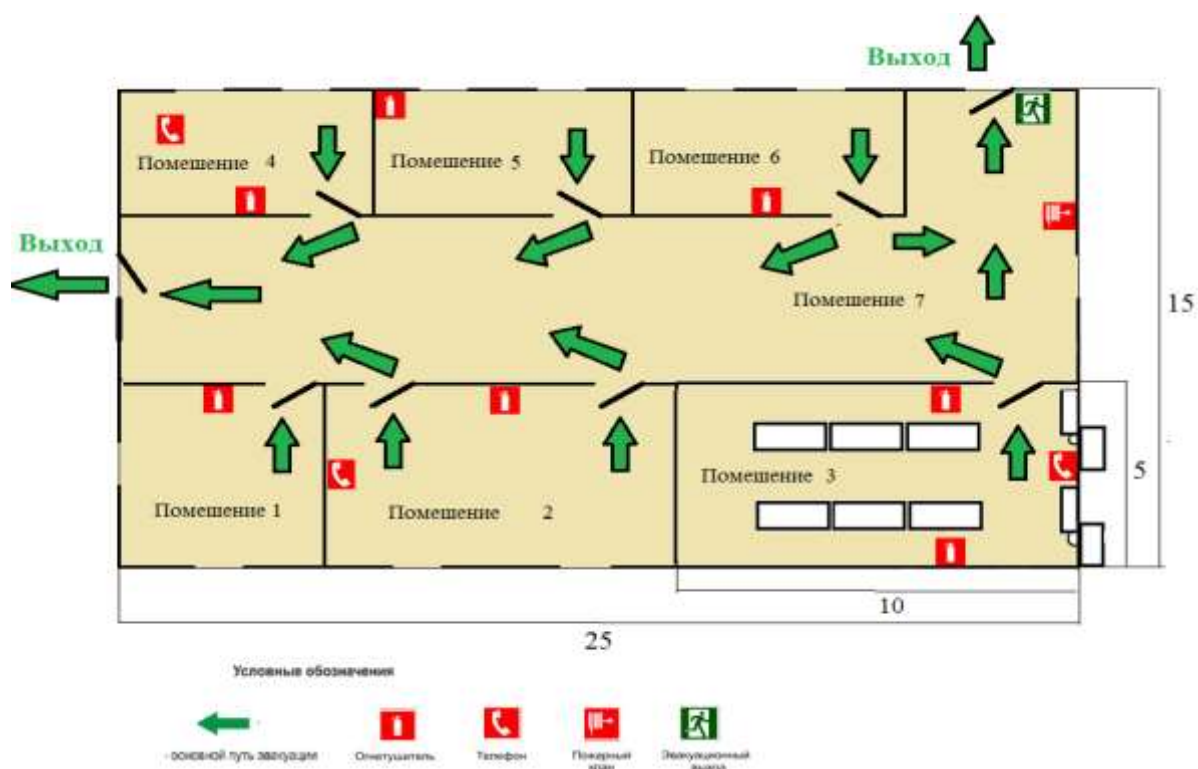


Рисунок 3.4 – План эвакуации и расположение огнетушителей на этаже

Выводы: в данном разделе дипломного проекта был произведен расчет условий труда для вычислительного центра, в том числе условия кондиционирования помещения в серверной. А также было рассчитано время эвакуации и количество огнетушителей, необходимые для пожарной безопасности.

4 Финансовый план

4.8 Расчет инвестиционных затрат

Капитальные вложение включают в себя стоимость оборудования, монтажных работ и транспортных услуг.

Общие капитальные вложения:

$$K = K_0 + K_M + K_{TP} \quad (4.1)$$

где K_0 – затраты на оборудование;

K_M - капитальные вложения на монтажные работы;

K_{TP} - капитальные вложения на транспортные расходы (5-10 % от стоимости оборудования).

Таблица 4.1 – Состав необходимого оборудования

Наименование технических средств	Количество	Цена	Стоимость
Компьютер на базе процессора Intel Xeon E5620 (4 Core, 12M Cache, 2.40 GHz)	1	70990	180990
Монитор Samsung 191n, диагональ 19",	1	37668	37668
Принтер HP LaserJet 1200	1	25880	25880
Источник бесперебойного питания ИБП 1000 Powerman	1	18248	18248
Электронный замок	1	5990	5990
Итого	5		268776

Теперь рассчитаем капитальные вложения на транспортные расходы (5% от общей стоимости оборудования K_0):

$$K_{TP} = K_0 \cdot 0,05 = 268776 \cdot 0,05 = 13439 \text{ тенге.} \quad (4.2)$$

Расходы за монтаж оборудования и налаживание пуска работы, производимые инженерами-монтажниками, так же составляют 5% от общей стоимости оборудования [19]

$$K_M = K_0 \cdot 0,05 = 268776 \cdot 0,05 = 13439 \text{ тенге.}$$

Таким образом, общие капитальные вложения К составили:

$$K = 268776 + 13439 + 13439 = 295654 \text{ тенге.}$$

Таблица 4.2 - Капитальные вложения

Наименование статей	Сумма, тг
Затраты на оборудование	268776
Капитальные вложения на монтажные работы	13439
Капитальные вложения на транспортные расходы	13439
Общие капитальные вложения	295654

4.9 Расчет эксплуатационных расходов

Разработка данной системы - сложный и трудоёмкий процесс, требующий наряду с интеллектуальными, техническими затратами и финансовых затрат. Поэтому необходимым является произведение расчёта себестоимости этой разработки.

Себестоимость выполнения научно исследовательских работы складывается из следующих статей затрат:

- а) заработная плата основных разработчиков (то есть кто непосредственно применяет участие в выполнении работ);
- б) дополнительную заработную плату;
- в) фонды оплат труда;
- г) отчисления из фонд оплат труда социального налога (11 %);
- ж) расходные материалы (бумага, картриджи, канцелярские принадлежности и другие);
- л) накладные расходы.

Таким образом, затраты или же эксплуатационных расходов на разработку данного программного комплекса определяются по формуле 4.3:

$$C = \text{ФОТ} + O_c + M + C_{\text{ПР}} + H \quad (4.3)$$

где ФОТ – фонды оплат труда;

O_c – социальный налог;

A – амортизационные отчисления;

Э – затраты на электроэнергию;

$C_{\text{ПР}}$ – прочие расходы;

H – накладные расходы;

Накладные расходы определяются в размере 25 % от всех затрат по формуле 7.4:

$$H = (\text{ФОТ} + O_c + A + \text{Э} + C_{\text{ПР}}) \cdot 0,25 \quad (4.4)$$

Определение фонды оплат труда ФОТ по формуле 7.5:

$$\text{ФОТ} = \text{З}_{\text{осн.}} + \text{З}_{\text{доп.}} \quad (4.5)$$

где $\text{З}_{\text{осн.}}$ – основную заработную плату;

$\text{З}_{\text{доп.}}$ – дополнительную заработную плату.

Дополнительная заработная плата (начисление премии) в среднем определяется в размере 10% от основной заработной платы, персональных надбавок:

$$\text{З}_{\text{доп.}} = \text{З}_{\text{осн.}} \cdot 0,1 \quad (4.6)$$

Пенсионные отчисления (ПО) определяются в размере 10% от ФОТ по формуле 4.7:

$$\text{ПО} = \text{ФОТ} \cdot 0,1 \quad (4.7)$$

Социальный налог составит 9,5 % от ФОТ за вычетом пенсионных отчислений.

$$\text{С}_H = (\text{ФОТ} - \text{ПО}) \cdot 0,095 \quad (4.8)$$

По существующему положению, в настоящее время, норма амортизации НА на оборудование связи составляет от 15 % до 40 % от стоимости всего оборудования, тогда амортизационные отчисления, A_i составят:

$$A_i = \frac{N_A \cdot C_{\text{ПЕР}} \cdot N}{100 \cdot 12 \cdot n} \quad (4.9)$$

где N_A - норма амортизации; A N

$C_{\text{ПЕР}}$ – первоначальная стоимость оборудования;

N - количество дней на выполнение работ.

Поскольку в процессе производства используется электрооборудование, то необходимо рассчитать затраты на электроэнергию. Затраты на электроэнергию для производственных нужд, включают в себя расходы электроэнергии на оборудование и дополнительные.

$$\text{Э} = \text{З}_{\text{эл.эн.обор.}} + \text{З}_{\text{доп.нуж}} \quad (4.10)$$

$$\text{З}_{\text{эл.эн.обор.}} = W \cdot T \cdot S \cdot K_{\text{исп}} \quad (4.11)$$

где W – потребляемая мощность, Вт;

T – время работы ($T=8760$ ч/год);

S – тариф ($1\text{ кВтч}=12,58\text{ тг}$);

$K_{\text{исп}}$ – коэффициент использования ($K_{\text{исп}} = 0,9$)

Затраты на дополнительные нужды возьмем по укрупненному показателю 5% от затрат на оборудование[19]:

$$З_{\text{доп.нуж.}} = 0,05 \cdot З_{\text{эл.эн.обор.}} \quad (4.12)$$

Количество и заработная плата работников, задействованных в разработке, указано в таблице 4.3.

Определим заработную плату каждого работника за один рабочий день – для этого месячный оклад работника делится на количество рабочих дней за прошедший период (это 22 дня – пятидневная рабочая неделя):

руководитель:

$$D = \frac{200000}{22} = 9091 \text{ тенге/день};$$

инженер:

$$D = \frac{150000}{22} = 6818 \text{ тенге/день};$$

Заработную плату за один час вычислим, разделив заработную плату работника за день на количество часов рабочего дня (8 часов):

руководитель:

$$D = \frac{9091}{8} = 1136 \text{ тенге/час};$$

инженер:

$$D = \frac{6818}{8} = 852 \text{ тенге/час};$$

Рассчитаем длительность циклов реализации проекта.

Длительность цикла в днях по каждому виду работ определяем по формуле:

$$t_n = \frac{T}{q_n \cdot z \cdot K} \quad (4.13)$$

где Т – трудоёмкость этапа, норма-час;

q_n – количество исполнителей по этапу;

z – продолжительность рабочего дня, z = 8 часов;

K – коэффициент выполнения норм времени, K = 1,1;

Полученную величину t_n округляем в большую сторону до целых дней.

Данные для расчетов возьмем из таблицы 4.3.

Таблица 4.3 – Расчет основной заработной платы персонала

Наименование этапов и содержание работ	Исполнитель	Трудоёмкость	Длительность цикла, дни	Заработная Плата за день работы, тенге	Сумма заработной платы, тенге
		Нормы			
Постановка задачи	Руководитель	16	2	9091	18182
Планирование ведения работы	Руководитель	24	3	9091	18182
	Инженеры (3 чел)	40	6	6818·3	122724
Разработка программы поиска людей	Инженеры (3 чел)	120	15	6818·3	306810
	ИТОГО:	200	26		465898

Определив заработную плату каждого работника за час работы по каждому наименованию проведённых работ, мы можем получить суммарную заработную плату, определенную как сумма оплаты труда всех работников, задействованных в разработке, по каждому наименованию проведённых работ (таблица 4.3).

$$З_{\text{осн}} = 465898 \text{ тенге};$$

Дополнительную заработную плату определим :

$$З_{\text{доп}} = 465898 \cdot 0,1 = 46589 \text{ тенге};$$

Вычислим фонд оплаты труда (ФОТ):

$$\text{ФОТ} = 465898 + 46589 = 512487 \text{ тенге};$$

Вычислим размер пенсионных отчислений:

$$\text{ПО} = 512487 \cdot 0,1 = 51248 \text{ тенге};$$

Вычислим размер отчислений на социальные нужды:

$$C_H = (512487 - 51248) \cdot 0,095 = 43818 \text{ тенге};$$

В том числе социальное страхование в размере

$$C_C = (512487 - 51248) \cdot 0,05 = 23062 \text{ тенге.}$$

4.10 Амортизационные отчисления, материальные затраты

Рассчитаем амортизационные отчисления по используемому оборудованию.

Нормы амортизационных отчислений для отрасли связи составляют до 25 % в год. Примем норму амортизации равной 15 %:

$$A_o = K \cdot \frac{H_A}{100\%} \quad (4.14)$$

$$A_o = 295654 \cdot \frac{15}{100\%} = 44348 \text{ тенге.}$$

Материальные затраты и расходы на запасные части и текущий ремонт составляют 0,5 % от капитальных вложений:

$$M = K \cdot \frac{0,5\%}{100\%} \quad (4.15)$$

$$M = 295654 \cdot 0,005 = 1478 \text{ тенге.}$$

4.11 Затраты на электроэнергию

Затраты на электроэнергию определим, как сумму затрат на основное оборудования:

$$З_{ЭЛ.ЭН.ОБОР.} = 0,9 \cdot 784 \cdot 9 \cdot 0,9 = 5715 \text{ тенге.}$$

Затраты на дополнительные нужды составят:

$$З_{доп.нуж.} = 0,05 \cdot 5715 = 286 \text{ тенге.}$$

Итого затраты на электроэнергию составляют :

$$\Xi = 5715 + 286 = 6001 \text{ тенге.}$$

Таблица 4.5 – Стоимость разработки по всем статьям затрат без учета накладных расходов

Наименование статей затрат	Сумма, тенге
ФОТ	512487
Отчисления на социальные нужды	43818
Амортизация	44348
Затраты на электроэнергию	6001
Материальные затраты	1478
ИТОГО:	608132

Теперь рассчитаем накладные расходы

$$H = 0,25 \cdot C \quad (4.16)$$

$$H = 0,25 \cdot 608132 = 152033 \text{ тенге.}$$

Сумма затрат вместе с накладными расходами составит 760165 тенге.

Себестоимость проекта составит сумму общих капитальных вложений и эксплуатационных расходов: 295654 тенге + 760165 тенге = 1 055819 тенге.



Рисунок 4.1 – Диаграмма структуры расходов

4.12 Доходы предприятия

Под доходами предприятия мы подразумеваем денежные средства, получаемые от предоставления определенной услуги или продукции. Данные средства служат для покрытия затрат и получения новой прибыли.

Была рассчитана себестоимость проекта, равная 1 055819 тенге.

Цена реализации программного продукта складывается из стоимости и чистого дохода:

$$Ц = С + П \quad (4.17)$$

где С – стоимость продукта;

П – чистый доход.

При определении первоначальной цены задаёмся желаемым уровнем рентабельности (40 %) для реализации программных продуктов:

$$Ц_{\Pi} = С \cdot \left(1 + \frac{P}{100}\right) \quad (4.18)$$

где Р – рентабельность (примем 40 %);

$$Ц_{\Pi} = 1\,055\,819 \cdot \left(1 + \frac{40}{100}\right) = 1\,478\,146 \text{ тенге.}$$

Определим цену реализации готового программного продукта:

$$Ц_{\text{Р}} = Ц_{\Pi} + \text{НДС} \quad (4.19)$$

Поскольку на сегодняшний день размер НДС в РК составляет 12 %, следовательно:

$$\text{НДС} = Ц_{\Pi} + 0,12 \quad (4.20)$$

$$\text{НДС} = 1\,478\,146 \cdot 0,12 = 177\,378 \text{ тенге,}$$

$$Ц_{\text{Р}} = 1\,478\,146 + 177\,378 = 1\,655\,524 \text{ тенге,}$$

Окончательная цена в условиях рыночных отношений будет зависеть и от спроса на интеллектуальный труд.

4.13 Расчет экономического эффекта, эффективной и срока окупаемости

Расчёт эксплуатационных затрат при использовании ПП

Годовые эксплуатационные текущие затраты в условиях функционирования ПП рассчитываются по формуле:

$$C = M + 3П + A + OT + НР = Ц_d(1 + НДС), \quad (4.8)$$

где M – годовые материальные затраты на сопровождение программного продукта, тг;

$3П$ – годовые затраты на оплату труда специалиста (программиста-разработчика), тг;

A – затраты на амортизацию, тг;

OT – отчисления на социальный налог, тг;

$НР$ – накладные расходы, тг.

Затраты на материалы в условиях функционирования ПП не ожидается, соответственно $M = 0$.

Годовые затраты на заработную плату сотрудников определяются по формуле:

$$3П = \frac{O_c \times Ч_c \times 12}{\Phi_{p.v.}} \times t_{общ} \times 12 \times (1 + K_d) \quad (4.9)$$

где O_c – оклад специалиста, тг/мес.;

$Ч_c$ – численность специалистов, участвующих в процессе, чел.;

$\Phi_{p.v.}$ – годовой фонд рабочего времени, час;

$t_{общ}$ – трудоёмкость решения задач в условиях функционирования ПО в месяц, час;

K_d – коэффициент дополнительной заработной платы.

Трудоёмкость решения задач вычисляется следующим образом:

$$t_{общ} = \sum_{k=1}^n t_k \times K_k, \quad (4.10)$$

где t_k – затраты времени на решение k -й задачи, час;

K_k – количество решаемых k -х задач в месяц, ед.

В ходе эксплуатации программы предполагается возникновение серверных ошибок. На поиск и исправление одной ошибки в среднем уходит около двух часов. В течение месяца эксплуатации ПО может возникать до 4 ошибок. Помимо этого, возможны некоторые изменения структуры серверного решения, либо доработка кода и программного обеспечения по

требованию заказчика. Всего компонентов в системе 4, на доработку одного уходит в среднем около 5 часов.

$$t_{\text{общ}} = 2 \times 4 + 5 \times 4 = 28 \text{ часов.}$$

Годовой фонд рабочего времени равен 1972 часам. Таким образом:

$$ЗП = \frac{200\,000 \times 1 \times 12}{1972} \times 28 \times 12 \times (1 + 0,1) = 449\,817 \text{ тенге}$$

Амортизационные отчисления выражаются в процентах к балансовой стоимости оборудования и вычисляются по формуле:

$$A = \sum_{i=1}^n C_{\text{обор.}i} \times H_{a_i}, \quad (4.11)$$

где $C_{\text{обор.}i}$ – первоначальная стоимость оборудования;

H_{a_i} – норма амортизации i -го оборудования;

i – вид оборудования;

n – количество видов оборудования.

Затраты на амортизацию составят:

$$A = 300\,000 \times 0,25 + 10\,000 \times 0,333 = 78\,330 \text{ тг.}$$

Расчёты отчислений на социальный налог и накладные расходы производятся так же, как и при разработке ПО.

$$ОТ = ЗП \times 10,46\% = 449\,817 \times 0,1046 = 47\,051 \text{ тенге.}$$

$$НР = ЗП \times \frac{H_{\text{нр}}}{100} = 449\,817 \times 0,7 = 314\,872 \text{ тенге.}$$

Все рассчитанные эксплуатационные затраты приводятся в таблице 4.7.

Таблица 4.7 – Эксплуатационные затраты

Вид затраты	Сумма, тг	В процентах от общей суммы, %
Материальные затраты (М)	0	0
Заработная плата (ЗП)	449 817	50,53
Амортизационные отчисления (А)	78 330	8,8
Социальные отчисления и налоги (ОТ)	47 051	5,29
Накладные расходы (НР)	314 872	35,38
Итого	890 070	100

Годовые эксплуатационные затраты в условиях функционирования ПО составляют 890 070 тенге.

4.14 Расчёт результатов от создания и использования ПО

Разрабатываемое программное решение предназначено для реализации технологии распознавания речи с использованием передовых архитектур ИНС.

Для анализа экономии в результате использования разрабатываемого ПП необходимо сравнить эксплуатационные расходы с внедрением ПП и без.

Затраты без использования ПП включают в себя:

- заработная плата специалиста, который осуществляет поддержку и сопровождение системы;
- износ оборудования;
- накладные расходы.

В организации имеются четыре сетевых администратора, заработная плата которых 170 000 тенге в месяц. Задачей каждого администратора является сетевой мониторинг работы распознавания речи, а также внесения необходимых изменений в топологию и инфраструктуру архитектуры нейронной сети.

Годовые затраты на оплату труда двух администраторов рассчитываются по формуле (4.9):

$$ЗП = \frac{170\,000 \cdot 2 \cdot 12}{1774} \cdot 24 \cdot 12 \cdot (1 + 0,1) = 728\,604 \text{ тенге.}$$

Социальные отчисления и налоги:

$$ОТ = 728\,604 \cdot 10,46\% = 76\,212 \text{ тенге.}$$

В серверной компании должен располагаться персональный компьютер с мощным процессором, здесь используется процессор Intel Xeon E5620 стоимостью 300 000 тг, еще серверном ПК установлена ОС ClearOS Home с годовой лицензией за 250 000 тг. Данное спецоборудование относят к материальным затратам.

Затраты на амортизацию с учетом серверного ПК и межсетевого экрана составят:

$$А = 300\,000 \times 0,25 + 10\,000 \times 0,333 + 300\,000 \times 0,2 = 138\,330 \text{ тенге.}$$

Накладные расходы составят:

$$НР = ЗП \cdot \frac{Н_{нр}}{100} = 728\,604 \times 0,7 = 510\,022 \text{ тенге.}$$

Сравним следующие статьи затрат без и с применением разрабатываемого ПО: заработная плата специалистов, расход на материалы и оборудование, износ оборудования и накладные расходы в таблице 4.8:

Таблица 4.8 – Сравнение статей затрат с использованием ПО и без

Статья затрат	Без использования ПП, тг	С использованием ПП, тг
Материальные затраты (М)	550 000	0
Заработная плата (ЗП)	728 604	449 817
Амортизационные отчисления (А)	138 330	78 330
Социальные отчисления и налоги (ОТ)	76 212	47 051
Накладные расходы (НР)	510 022	314 872
Итого	2 003 168	890 070

Вычислим ожидаемую условно-годовую экономию по формуле:

$$\mathcal{E}_{\text{уг}} = C_1 - C_2 + \sum \mathcal{E}_i, \quad (4.12)$$

где $\mathcal{E}_{\text{уг}}$ – величина экономии, тг;

C_1 и C_2 – показатели затрат по базовому и внедряемому варианту, тг;

$\sum \mathcal{E}_i$ – ожидаемый дополнительный эффект от различных факторов, тг.

Таким образом, ожидаемая условно-годовая экономия составит:

$$\mathcal{E}_{\text{уг}} = 2\,003\,168 - 890\,070 = 1\,113\,098 \text{ тг}$$

4.15 Расчёт основных показателей экономической эффективности

Эффективность разрабатываемого ПО формируется за счёт экономии в сравнении с предыдущим периодом работы без его использования.

Для расчёта величины ожидаемого годового экономического эффекта от внедрения ПО используется формула:

$$\mathcal{E}_{\text{г}} = \mathcal{E}_{\text{уг}} - K * E_{\text{н}}, \quad (4.13)$$

где $\mathcal{E}_{\text{г}}$ – ожидаемый годовой экономический эффект, тг;

$\mathcal{E}_{\text{уг}}$ – ожидаемая условно-годовая экономия, тг;

K – капитальные вложения, тг;

$E_{\text{н}}$ – нормативный коэффициент экономической эффективности капитальных вложений.

Нормативный коэффициент экономической эффективности капитальных вложений вычисляется по формуле:

$$E_H = \frac{1}{T_H}, \quad (4.14)$$

где T_H – нормативный срок окупаемости капитальных вложений, лет.
Для ПО примем равным четырём годам ($T_H = 4$)

$$E_H = \frac{1}{4} = 0,25.$$

Ожидаемый годовой экономический эффект согласно формуле (4.13) равен:

$$\mathcal{E}_r = 1\,113\,098 - 1655524 * 0,25 = 768\,785 \text{ тенге.}$$

Расчетный коэффициент экономической эффективности капитальных вложений находится по формуле:

$$E_p = \frac{\mathcal{E}_r}{K}, \quad (4.15)$$

где E_p – расчётный коэффициент экономической эффективности капитальных вложений;

\mathcal{E}_r – ожидаемая годовая экономия, тг;

K — капитальные вложения на создание системы, тг.

$$E_p = \frac{1\,113\,098}{1\,655\,524} = 0,67$$

Расчетный срок окупаемости капитальных вложений составляет:

$$T_p = \frac{1}{E_p} = \frac{1}{0,67} = 1,49 \text{ года} \approx 17,9 \text{ месяцев.}$$

Основные показатели экономической эффективности проекта представлены в таблице 4.9:

Таблица 4.9 – Основные показатели экономической эффективности

Показатель	Значение
Ожидаемый годовой экономический эффект, тг	1 113 098
Коэффициент экономической эффективности капитальных вложений	0,67
Срок окупаемости капитальных вложений, месяцев	17,9

Выводы по технико-экономическому обоснованию

В экономической части были рассчитаны и проанализированы возможные затраты на внедрение программного продукта. Развертываемое ПО позволит реализовать распознавание речи с использованием передовых архитектур ИНС. для небольшой организации за относительно невысокую

стоимость, т.к. все программное обеспечение не включается в состав затрат, благодаря лицензии на передовых архитектур ИНС, что означает о свободном распространение данного ПО.

На оплату труда сотрудника уходит большая часть затрат, а именно 55,56%, далее следует 39,42% затрат на приобретение спецоборудования. и остальные, уже менее значительные затраты.

Последними этапами были расчеты эксплуатационных затрат используемого ПО и расчеты показателей экономической эффективности, по результатам которых, годовые эксплуатационные затраты в условиях функционирования распознавания речи составляют 890 070 тенге, что можно считать вполне разумным значением ввиду неизменной серверной инфраструктуры оборудования, условно-годовая экономия составит 1 113 098, а срок окупаемости 17,9 месяцев.

Заключение

В ходе выполнения данной дипломной работы были рассмотрены нейросети, их виды, строение нейронной сети. рассмотрены основные теоретические основы нейронных сетей: их классификация, принцип работы. Рассмотрены сами принципы действия искусственных нейронов и распознавания речи методы определения ошибок, обучение. Было реализовано два способа сегментация речи, первое с помощью функций и написания кода в MatLab . Второе это реализация сегментации в среде Simulink с использованием блока VAD. Так же было смоделирована работа сети Кохонена в программе MatLab. А точнее моделирование слоя и самоорганизующийся карты Кохонена. И предоставлена реализация технологии на языке программирования Python. В итоге нейросеть прошла обучение и смогла определить речь. Так же в разделе безопасности жизнедеятельности было рассчитана кондиционирование и пожарная безопасность, и еще разработан финансовый план. В кондиционирование и пожарная безопасность.

Список литературы

- 1 Кэмпбелл Дж. П. Распознавание спикера: Учебное пособие // Труды IEEE. - 1997. - № 9. - С. 1437-1462.
- 2 Методы автоматического распознавания речи [пер. с англ.] [Текст] / Под ред. У. Ли. - М.: Мир, 1983. - Т.1. - 328 с.; - Т.2. - 392 с.
- 3 Доклад Томаса Зенкель. Языковое моделирование на основе классов с помощью нейронных сетей [пер. с нем.], Кáрлсруэ 2015 г;
- 4 История возникновения нейронных сетей URL – <https://neuronus.com/history/5-istoriya-nejronnykh-setej.html> (дата обращения 20.05.2021)
- 5 К.В. Воронцов. Прогресс нейронных сетей URL – <https://postnauka.ru/talks/80077> (дата обращения 25.05.2021)
- 6 С.П. Хабаров .Интеллектуальные информационные системы URL – <https://docplayer.ru/30594591-11-neyronnye-seti-raspoznavanie-obrazov.html> (9.10.2018)
- 7 Клименко Н.С. Разработка структуры текстонезависимой системы идентификации диктора // Искусственный интеллект. – 2012. – №4. – С. 161–171.
- 8 Начальник П. Речевые Команды: общедоступный набор данных для распознавания речи однословного, Санкт-Петербург, 2017.
- 9 Кохонен Т. Самоорганизация и ассоциативная память. Берлин , 1987.
- 10 М. В. Бураков. Нейронные сети и нейроконтроллеры - Учебное пособие. Санкт-Петербург, 2013.
- 11 Теуво М. Кохонен. Самоорганизующиеся карты. Издательство БИНОМ. Лаборатория знаний: 2014 С.: 659 ISBN: 978-5-99-63-1348-8
- 12 А.А. Байзаков, А.С. Бегимбетова, М.К. Дюсебаев, Т.С. Санатова. Охрана труда. Методические указания к выполнению лабораторных работ (для студентов всех специальностей очно-заочной формы обучения). – Алматы: АИЭС, 2004.
- 13 ГОСТ 30494-96 Здания жилые и общественные. Параметры безопасности в помещениях. – М.: Радио и связь, 1989.
- 14 Шокина Л.Г. Охрана труда на предприятиях связи, учебник, - СПб.: БХВ-Петербург 1983 г.
- 15 МУ Хакимжанов Т.Е. Расчет аспирационных систем. Дипломное проектирование. Для студентов всех форм обучения всех специальностей. – Алматы: АИЭС, 2002. – 30 с.
- 16 Абикенова А.А., Санатова Т.С . МУ Методические указания к выполнению раздела «Пожарная профилактика» в выпускных работах для всех специальностей – Алматы: АИЭС, 2009.
- 17 Н.Г. Приходько, Ф.Р. Жандаулетова. МУ "Основы пожарной безопасности" – АУЭС 2013.