БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

На правах рукописи
УДК 004.855.5

Кунцевич Сергей Казимирович

Алгоритм распознавания эмоций на основе LSTM-сетей глубокого обучения

Реферат по
«Основам информационных технологий»

Магистранта кафедры радиофизики и цифровых медиатехнологий

Факультет радиофизики и компьютерных технологий

Специальность: 1-31 80 07 Радиофизика

Рецензент:
_____

Минск, 2017

ОГЛАВЛЕНИЕ

ПЕРЕЧЕНЬ УСЛОВНЫХ ОБОЗНАЧЕНИЙ

CNN – Convolutional Neural Network (свёрточная нейронная сеть)

RNN– Recurrent Neural Network (рекуррентная нейронная сеть)

LSTM – long-short term memory(сеть долго-краткосрочной памяти)

ВВЕДЕНИЕ

Несмотря на большой прогресс, достигнутый в области искусственного интеллекта, возможность естественного взаимодействия с машинами на данный момент не реализована, отчасти потому, что машины не понимают наших эмоций. В настоящее время задача распознавания речевых эмоций становится все более актуальной.

Распознавание эмоций речи является сложной задачей, так как не существует однозначно эффективного алгоритма для решения данной задачи. Осложняет задачу так же тот факт, что нет достаточно строгого определения, что такое эмоция. Кроме того разные люди по-разному классифицируют эмоциональную речь.

Целью данной дипломной работы является разработка алгоритма распознавания эмоций речи на основе искусственных нейронных LSTM-сетей глубокого обучения.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Рассмотреть основные принципы и существующие методы решения задач машинного обучения, распознавания образов и компьютерного зрения.
2. Изучить технологии построения и обучения нейронных сетей глубокого обучения с использованием графических видеоадаптеров.
3. Разработать алгоритм распознавания эмоций речи с использованием LSTM-сетей глубокого обучения.
4. Обучить LSTM-классификатор провести экспериментальное исследование разработанного алгоритма распознавания эмоций.

ГЛАВА 1

ЭМОЦИИ И СПОСОБЫ ИХ ОПИСАНИЯ

1.1 Модели эмоциональных процессов

Первые попытки определения эмоциональных состояний с использованием статистических закономерностей, присущих определенным акустическим особенностям голоса, были предприняты в середине 80-х годов [1, 2]. В дальнейшем интерес к задаче автоматического распознавания эмоций по голосу только возрастал. Количество ежегодно публикуемых статей, относящихся к данной тематике, с середины 90-х годов до настоящего момента, возросло более чем в 10 раз, а эффективность распознавания приблизилась к человеческой.

Однако несмотря на достигнутые успехи в сфере голосового детектирования эмоциональных состояний, проблема ещё очень далека от своего окончательного решения. Существует ряд серьёзных препятствий, значительно усложняющих разработку эффективных систем распознавания эмоционального состояния:

Нет чёткого определения понятия «эмоция». При любых попытках его формализации мы натыкаемся на многообразие психологических моделей эмоциональных процессов, демонстрирующих различные ракурсы восприятия и модели описания эмоционального состояния диктора;

1) Отсутствие единой теории, связывающей внутренние состояния диктора с особенностями его речи. Несмотря на достигнутые в этой сфере успехи, общепринятого подхода пока не существует [3].

Способы выражения эмоций могут варьироваться для разных дикторов и в разных культурных контекстах. Последнее играет важную роль при разработке языконезависимых систем распознавания эмоционального состояния.

1.2 Основные парадигмы исследования эмоций

В психологии выделяются четыре основных направления в определении эмоций, для каждого из которых характерен свой набор базовых предположений и допущений.

Согласно эволюционному подходу, основоположником которого является Чарльз Дарвин, эмоции являются паттернами реагирования, сформировавшимися в процессе эволюции под действием естественного отбора. Соответственно, эмоции рассматриваются как общие для всех людей и даже некоторых приматов. В рамках эволюционного подхода возникла концепция базовых эмоций, сформировавшихся в ходе эволюции и способствующих выживанию вида.

Важнейшим открытием, сделанным в рамках эволюционного подхода, является универсальность лицевой экспрессии, продемонстрированная Полом Экманом. Он показал, что, по крайней мере, шесть эмоций (счастье, печаль, гнев, страх, удивление и раздражение) выражаются и распознаются на лице одинаковым образом во множестве различных культур. В то же время, Экман отметил и специфичные для различных культур правила экспрессии, регулирующие в какой ситуации и как некоторые эмоции можно демонстрировать, а некоторые – нет. Тем не менее, именно исследования универсальности эмоциональной экспрессии обосновывают возможность создания языконезависимых систем распознавания эмоционального состояния.

В подходе, основателем которого был Уильям Джеймс, важнейшая роль в переживании эмоций отводится восприятию изменений в теле. Как и в подходе Дарвина, телесные изменения в ответ на некоторые внешние стимулы происходят более-менее автоматически, а эмоции возникают как восприятие этих изменений.

В когнитивном подходе к эмоциональным процессам центральной является концепция оценки, производимой в ходе относительно низкоуровневых автоматических когнитивных процессов. Оценка внешнего

стимула определяет его значимость для индивида и запускает эмоциональный процесс в качестве подходящей реакции.

Из множества исследований, проведённых в рамках когнитивного подхода, наибольшего внимания разработчиков систем голосового детектирования эмоций заслуживают работы Клауса Шерера. Это обусловлено тем фактом, что, исходя из своей компонентной модели эмоциональных процессов, Шерер сделал детальные предсказания изменений голоса, соответствующих различным эмоциональным состояниям, большая часть которых позднее нашла экспериментальное подтверждение.

Парадигма социального конструктивизма рассматривает эмоции как сконструированные социумом паттерны. Они выполняют социальные задачи, регулируя различными способами взаимодействия между людьми. В рамках этого подхода, не только эмоциональная экспрессия, но и сами эмоции, включая субъективный опыт, рассматриваются как сконструированные социумом. Биологические основания эмоциональных процессов признаются, однако рассматриваются как второстепенные по отношению к социально выстроенным механизмам.

На первый взгляд, вышеописанные подходы противоречат один другому. К примеру, центральная для подходов Дарвина и Джеймса идея о существовании универсальных «базовых» эмоций несовместима с центральной концепцией социального конструктивизма, гласящей, что эмоции обусловлены социально сконструированными «сценариями». Однако при более внимательном рассмотрении, выясняется, что эти подходы просто уделяют наибольшее внимание различным аспектам эмоциональных процессов.

1.3 Подходы к описанию эмоциональных процессов

В рамках некоторых из описанных в предыдущем разделе подходов (в первую очередь, в подходах Дарвина и Джеймса) возникает мысль о том, что определённую часть «проявленных» эмоций можно определить как базовые, а

остальные являются вторичными, производными от базовых. С позиций дарвинизма, базовые эмоции соответствуют специфическим, эволюционно обусловленным функциям. Соответственно, ожидается, что эти эмоции универсальны, и могут быть обнаружены у всех людей. Дополнением Джеймса к этой концепции является предположение о наличии специфических физиологических паттернов и, вероятно, целых эмоциональных подсистем в организме.

Для идентификации базовых эмоций был применён ряд подходов, среди них – эволюционный, нейрофизиологический, психоаналитический, автономный, подход, использующий лицевую экспрессию, эмпирическую классификацию и экспериментальный. Различие подходов приводит в итоге к различным наборам базовых эмоций, выделяемым в процессе проведения исследований (таблица 1.1).

Таблица 1.1 Наборы базовых эмоций, выделенных различными исследователями

Авторы	Базовые эмоции
R. Plutchik, 1962 [4]	одобрение, злость, предвкушение, отвращение, радость, страх, печаль, удивление
P. Ekman et al., 1982 [5]	злость, отвращение, страх, радость, печаль, удивление
K. Oatley and P.N. Johnson-Laird, 1987 [6]	злость, отвращение, беспокойство (тревога), счастье, печаль
B. Weiner and S. Graham, 1984 [7]	счастье, печаль

Модели, опирающиеся на выделение эмоциональных категорий, предположительно являющиеся базовыми по отношению к другим, в литературе часто называются «дискретными» либо «палитровыми». Альтернативой им являются так называемые «непрерывные» модели эмоциональных процессов, вводящие в рассмотрение набор непрерывных шкал в эмоциональном пространстве.

Впервые идея о существовании небольшого числа базовых эмоций была высказана Вильгельмом Вундом в 1896 году. В 1957 году Осгуд с сотрудниками применил метод семантического дифференциала для идентификации эмоциональных шкал. В ходе анализа три фактора были выделены им в качестве фундаментальных для классификации эмоций – оценка, мощность и активность. Позднее был представлен ряд свидетельств в пользу существования трех шкал, организующих эмоциональное пространство (рисунок 1.1). Шкалы были соотнесены с концепциями удовольствия, активности и доминирования.

Рисунок 1.1 Непрерывная шкала эмоций

В двумерном приближении, сохраняющем основные свойства эмоционального пространства, шкалы однозначно интерпретировались как удовольствие и активность. Таким образом, непрерывная шкала хорошо приспособлена для картирования человеческих эмоций. Она выражает свойства эмоций, субъективно оцениваемые как наиболее значимые.

ГЛАВА 2

СОВРЕМЕННЫЕ МЕТОДЫ МАШИННОГО ОБУЧЕНИЯ

2.1 Основные сведения о машинном обучении

Для решения задачи распознавания эмоций в настоящие время используются методы машинного обучения.

Машинное обучение – это обширный подраздел теории искусственного интеллекта, изучающий методы построения алгоритмов, способных обучаться.

Различают два типа машинного обучения: обучение по прецедентам (индуктивное обучение), основанное на эмпирических данных, и дедуктивное обучение, предполагающее формализацию знаний и формирование базы знаний. Дедуктивное обучение предполагает формализацию знаний экспертов и их перенос в компьютер в виде базы знаний. Дедуктивное обучение принято относить к области экспертных систем, поэтому термины машинное обучение и обучение по прецедентам можно считать синонимами.

Раздел машинного обучения возник в результате разделения науки о нейронных сетях в рамках науки об искусственном интеллекте на методы обучения сетей и виды топологий архитектуры сетей, вобрав в себя некоторые другие области, такие как методы математической статистики и теорию дискретного анализа [17]. Этим обусловлена специфика рассматриваемых способов обучения в рамках дисциплины:

• Обучение с учителем – для каждого прецедента существует пара «ситуация, решение»

• Обучение без учителя – система группирует объекты в кластеры и понижает размерность входной информации, используя данные о попарном сходстве

• Обучение с подкреплением – для каждого прецедента существует пара «ситуация, реакция среды» (обучение с подкреплением можно считать частным

случаем обучения с учителем, так и частным случаем обучения без учителя)

Существуют также другие, менее распространённые способы обучения, например, активное обучение(обучаемый алгоритм имеет возможность назначить следующую исследуемую ситуацию), частичное привлечение учителя, трансдуктивное обучение, многозадачное обучение, многовариантное обучение и другие, но отличия в этих способах несущественны в рамках рассматриваемой проблемы. Все перечисленные способы можно использовать для классификации традиционных методов машинного обучения и для классификации алгоритмов обучения нейронных сетей, реализующих любой из них. Современное машинное обучение сталкивается с острой проблемой универсальности, поскольку практически не существует однородного пространства алгоритмов и метода общего решения проблемы индукции [18].

Теория распознавания образов.

Теория распознавания образов существует как раздел информатики и смежных дисциплин, развивающий методы классификации и идентификации объектов различной природы: сигналов, ситуаций, предметов, характеризуемых конечным множеством некоторых признаков [19].

Проблема распознавания образов также оказывается в поле междисциплинарных исследований – в том числе в связи с работой по созданию искусственного интеллекта, а также часто используется при решении практических задач области компьютерного зрения, к чему относится и рассматриваемый случай.

При постановке классической задачи распознавания образов принято использовать строгий математический язык, основываясь на логических рассуждениях и математических доказательствах. В противоположность этому подходу, существуют методы распознавания образов с использованием машинного обучения и искусственных нейронных сетей, сформированные не столь строго формализованными подходами к распознаванию, но, как будет показано далее, демонстрирующие не худший, а в некоторых случаях

значительно превосходящий классические методы результат.

Характер проблематики распознавания образов также ограничивает область применения различных традиционных методов узкими специализированными направлениями, в каждом из которых наиболее эффективными оказываются одни методы и неэффективными другие. Этот фактор обуславливает сложность общего решения проблемы индукции и в этой дисциплине.

Компьютерное зрение.

Область компьютерного зрения, в свою очередь,развивается как теория и технология создания машин,производящих обнаружение, отслеживание и классификацию объектов. Как научная дисциплина, компьютерное зрение тесно связано с машинным обучением и теорией распознавания образов, но относится к более специализированной сфере теории и технологии создания искусственных систем, получающих информацию и оперирующих информацией из изображений [20]. Автоматическое планирование или принятие решений на основе подсистем компьютерного зрения так же занимает важную часть в области искусственного интеллекта, поскольку автономные системы достаточно сложного уровня организации, выполняющие некоторые механические действия (например, перемещение робота через некоторую среду), нуждаются в высокоуровневых данных, представляющих информацию о среде, в которой они функционируют. Поэтому компьютерное зрение, как и распознавание образов, тесно связано с обработкой сигналов, ведь многие методы обработки одномерных сигналов могут быть естественным путём расширены для обработки двумерных или многомерных сигналов в рамках теории компьютерного зрения, использующей статистику, методы оптимизации и геометрии.

Компьютерное зрение также тесно связано с областями обработки изображений и машинного зрения. Область обработки изображений сосредоточена на анализе и преобразованиях одних изображений в другие с

использованием математических методов. Область машинного зрения, часто рассматриваемая как раздел более общей теории компьютерного зрения, сосредоточена на технологиях промышленного применения. Существуют также и другие более узкие области, связанные, зависящие или возникшие на основе теории компьютерного зрения, такие как, например, область визуализации, сосредоточенная на процессе создания изображений, их обработкой и анализом [20].

В данный момент не существует ни стандартной формулировки проблемы компьютерного зрения, ни формулировки того как должна решаться проблема компьютерного зрения, вместо чего разработана масса методов для решения различных строго определённых задач,при этом используемые методы редко обобщаются для широкого круга применения.

2.2 Основные сведения об искусственных нейронная сетях

Искусственная нейронная сеть является концептуальной моделью биологической нейронной сети и состоит из связанных различным образом слоёв искусственных нейронов, организующих общую активную структуру и функционально влияющих на работу друг друга. В большинстве архитектур искусственных нейронных сетей активность нейрона определяется преобразованием внешнего суммарного воздействия других нейронов на данный нейрон[21].

С момента своего зарождения технологии искусственных нейронных сетей развивались достаточно обособленным от классических методов путём, нередко в корне меняя представление о предмете в совокупной проблематике теорий машинного обучения и распознавания образов, оказывая значительное влияние на теоретический, терминологический и методологический аппараты этих дисциплин. С этого времени в научном сообществе произошло несколько спадов и подъёмов интереса к этому направлению, но, благодаря некоторым прорывам в теории искусственных нейронных сетей, широкое практическое

применение нейросетевые технологии получили сравнительно недавно.

Спустя некоторое время после развития базовых моделей искусственных нейронных сетей, произошло принципиальное разделение объемлющей науки о нейросетях на виды топологий архитектуры сетей и методы обучения сетей.

2.2.1 Архитектура искусственной нейронной сети

Существуют различные классификации искусственных нейронных сетей по ряду признаков. По одному из топологических признаков искусственные нейронные сети можно классифицировать как:

• Полносвязные искусственные нейронные сети – каждый нейрон связан с остальными нейронами в сети, включая себя самого (рисунок 2.1, а)

• Многослойные искусственные нейронные сети – нейроны объединяются в слои, нейроны предыдущего слоя связаны с нейронами следующего слоя (рисунок 2.1, б)

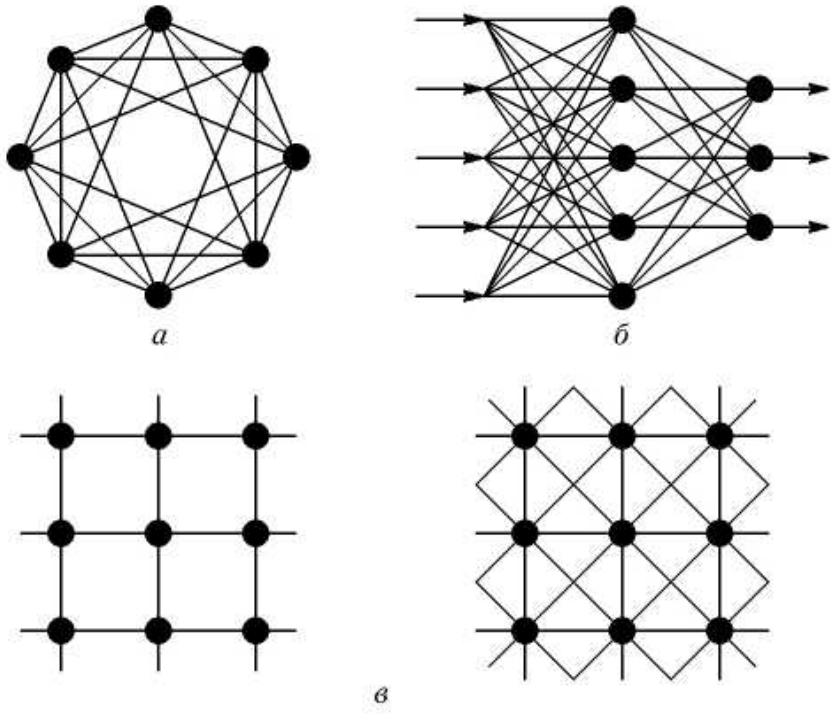


Рисунок 2.1 Архитектуры нейронных сетей: а – полносвязная искусственная нейронная сеть, б – многослойная искусственная нейронная сеть с последовательными связями, в – слабосвязные искусственные нейронные сети

2.2.2 Виды многослойных искусственных нейронных сетей

Различия вычислительных процессов в нейронных сетях часто обусловлены способом взаимосвязей нейронов. По совокупности критериев на сегодняшний день многослойные архитектуры искусственных нейронных сетей можно разделить на статические и динамические. Каждый из классов архитектур нейронных сетей может включать множество подклассов, реализуя различные подходы, ниже будут приведены основные из них.

К статическим архитектурам относят сети прямого распространения, в которых реализована однонаправленная связь между слоями, отсутствуют динамические элементы и обратная связь, а выход обученной искусственной нейронной сети однозначно определяется входом и не зависит от предыдущих состояний сети [22].

Статические искусственные нейронные сети прямого распространения:

• Перцептрон

• Нейронная сеть Кохонена

• Когнитрон

• Неокогнитрон

• Свёрточная нейронная сеть

В противоположность статическим архитектурам,существуют динамические архитектуры искусственных нейронных сеть, реализующие рекуррентную структуру с использованием обратных связей, благодаря чему состояние сети в каждый момент времени зависит от предшествующего состояния. Рекуррентные нейронные сети как правило базируются на многослойном перцептроне.

Динамические рекуррентные нейронные сети с обратными связями:

• Нейронная сеть Хопфилда

• Нейронная сеть Коско

• Нейронная сеть Джордана

• Нейронная сеть Элмана

2.2.3 Функции активации нейронов

Для реализации нелинейности при активации нейрона, его активность, помимо различных видов сумматоров и систем весов на входах, определяется функцией одного аргумента – функцией активации. Нейрон в целом реализует скалярную функцию векторного аргумента, а выходной сигнал нейрона определяется видом функции активации и может быть действительным или целым. Функция активации применяется к взвешенной сумме постсинаптических сигналов на входе нейрона. Таким образом, активность нейрона полностью определяется его параметрами – весами и его функцией активации. Существует множество передаточных функций,применяемых на практике использования нейронных сетей, некоторые из них служат для реализации нелинейности системы. Выбор той или иной функции активации часто зависит от условий задачи и структуры сети. Некоторые из рассматриваемых передаточных функций применяются только в устаревающих системах или в целях обучения, но считаются классическими и упоминаются всякий раз при изучении искусственных нейронных сетей.

2.2.4 Обучение нейронных сетей

Процесс обучения нейронных сетей рассматривается как настройка архитектуры и весов связей между нейронами(параметров) для эффективного выполнения поставленных перед искусственной нейронной сетью задач. Существует два обширных класса обучения нейронных сетей: класс детерминированных методов и класс стохастических методов.

В класс детерминированных методов входят методы, в основе которых лежит итеративная коррекция параметров сети, в ходе текущей итерации основывающаяся на текущих параметрах. Основным детерминированным методом и самым распространённым методом обучения нейронной сети сегодня вообще является метод обратного распространения ошибки [18].

В класс стохастических методов входят методы,изменяющие параметры

сети случайным образом и сохраняющие только те изменения параметров, которые привели к улучшению результатов. Стохастические алгоритмы обучения реализуются с помощью сравнения ошибок и некоторые из них связаны с проблемой «ловушки локального минимума», решаемой с помощью некоторых усложнений стохастических алгоритмов.

ГЛАВА 3

РАЗРАБОТКА АЛГОРИТМА РАСПОЗНАВАНИЯ ЭМОЦИЙ РЕЧИ НА ОСНОВЕ LSTM-СЕТЕЙ ГЛУБОКОГО ОБУЧЕНИЯ

Традиционные нейронные сети не имеют обратной связи, т.е. на каждом новом уровне не учитывают то, что происходило до этого, что является их основным недостатком. Например, необходимо классифицировать события кинофильма, происходящие в каждый момент времени.

Рекуррентные нейронные сети решают эту проблему. Одной из ключевых особенностей рекуррентных нейронных сетей является то, что они способны использовать информацию о предыдущем кадре видео для обработке текущего кадра (рисунок 3.1).

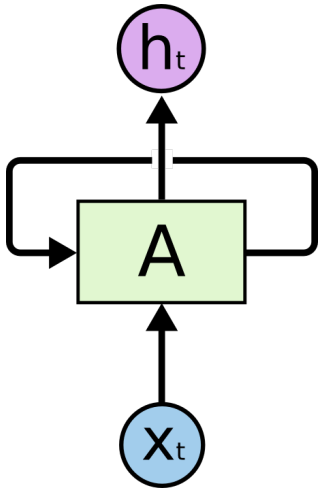


Рисунок 3.1 Схематическое представление рекуррентной нейронной сети

Рекуррентные нейронные сети имеют форму цепи повторяющих модулей (repeating module) нейронной сети. Однако в стандартной рекуррентной нейронной сети эти повторяющие модули будут иметь очень простую структуру, например, всего один слой гиперболического тангенса (tanh).

LSTM-сети специально спроектированы таким образом, чтобы избежать проблемы долгосрочных зависимостей.

3.1 Основные сведения о LSTM-сетях

Сети долго-краткосрочной памяти (Long Short Term Memory) - обычно просто называют “LSTM” - особый вид рекуррентной нейронной сети, способных к обучению долгосрочным зависимостям. Они были предложены Хохрейтером и Шмидхубером и доработаны и популяризованы другими в последующей работе. Они работают невероятно хорошо на большом разнообразии проблем и в данный момент широко применяются.

Повторяющий модуль в стандартной рекуррентной нейронной сети содержит один слой (рисунок 3.2). LSTM тоже имеют такую цепную структуру, но повторяющий модуль имеет другое строение. Вместо одного нейронного слоя их четыре, причём они взаимодействуют особым образом (рисунок 3.3).

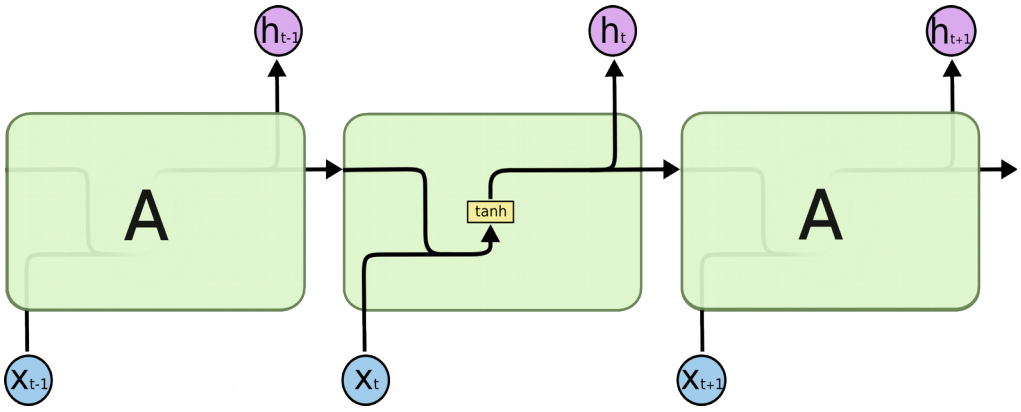


Рисунок 3.2 Повторяющий модуль в стандартной рекуррентной нейронной сети

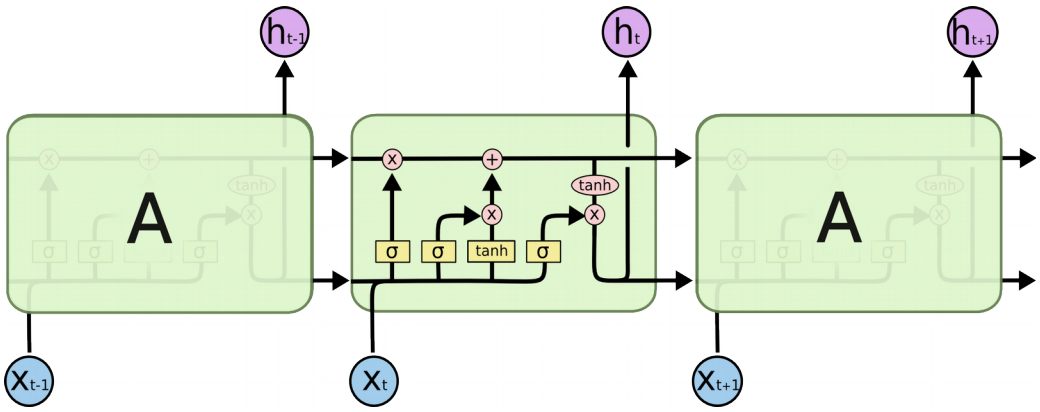


Рисунок 3.3 Повторяющий модуль в LSTM

В диаграмме выше каждая линия передаёт целый вектор от выхода

одного узла к входам других. Круги представляют поточечные операторы, такие как сложение векторов, в то время, как прямоугольники - это обученные слои нейронной сети. Сливающиеся линии обозначают конкатенацию, в то время как ветвящиеся линии обозначают, что их содержимое копируется, и копии отправляются в разные места (рисунок 3.4).

Рисунок 3.4 Условные обозначения

Ключ к LSTM — клеточное состояние (cell state) - горизонтальная линия, проходящая сквозь верхнюю часть диаграммы (рисунок 3.5).

Клеточное состояние — можно представить в виде ленты конвейера. Она движется прямо вдоль всей цепи только лишь с небольшими линейными взаимодействиями. Информация может просто течь по ней без изменений.

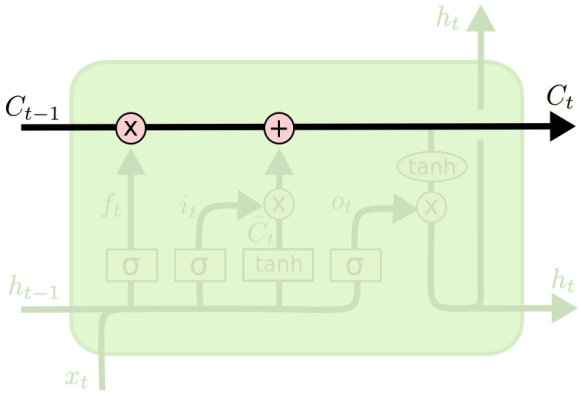


Рисунок 3.5 Клеточное состояние

LSTM имеет способность удалять или добавлять информацию к клеточному состоянию, однако эта способность тщательно регулируется структурами, называемыми вентилями (gates).

Вентили - это способ избирательно пропускать информацию. Они

составлены из сигмоидного слоя нейронной сети и операции поточечного умножения (pointwise multiplication) (рисунок 3.5).

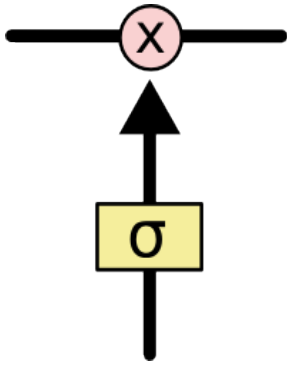


Рисунок 3.6. Схема вентиля

Сигмоидный слой подает на выход числа между нулем и единицей, описывая таким образом, насколько каждый компонент должен быть пропущен сквозь вентиль. Ноль - “ничего не пропускать”, один - “пропускать все”.

LSTM имеет три таких вентиля, чтобы защищать и контролировать клеточное состояние.

3.2 Основные сведения о свёрточных нейронный сетях и использующихся слоях

В современных искусственных нейронных сетях для моделирования вероятностного распределения используются:

- softmax-слои, включающие N нейронов по целевому количеству классов. Выход каждого нейрона зависит от сумматоров всех остальных нейронов слоя.
- Max-Pooling-слои или субдискретизирующие (subsampling) слои выполняющие уменьшение входной карты признаков. Чаще всего для этого используется метод выбора максимального элемента (maxpooling). Использование max-pooling позволяет сделать сеть инвариантной к масштабным преобразованиям.

3.2.1 Архитектура сверточной нейронной сети LeNet

LeNet - первая современная свёрточная нейросеть, разработанная в 80-90-х годах Яном ЛеКуном, обучалась для распознавания рукописных символов. Основным отличием от неокогнитрона было включение субдискретизирующего max-pooling слоя после каждого свёрточного слоя и включение полносвязных слоёв на выходе сети (рисунок 3.7). С этого момента свёрточные и max-pooling-слои становятся сердцем современных глубоких сетей. Эта разработка стала стандартом среди свёрточных нейронных сетей, а рассматриваемый в статье набор рукописных цифр MNIST (Mixed National Institute of Standards and Technology database) и по сей день используется для сравнительной оценки качества классификации алгоритмов в области обработки изображений.

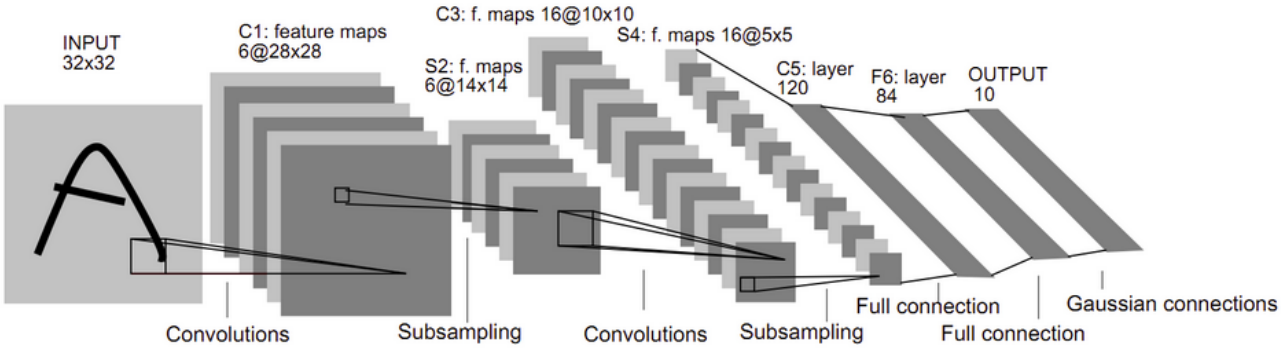


Рисунок 3.7 Архитектура LeNet

3.2.2 Архитектура сверточной нейронной сети GoogLeNet

В 2014 году ImageNet Recognition Challenge выиграла нейросеть Google GoogLeNet, на основе первой Inception-архитектуры. В архитектуре был уменьшен размер свёртки, были параллельно включены небольшие свёртки разного масштаба, удалены полносвязные слои на выходе сети и вместо них включены слои, названные global average pooling. Новая архитектура получила активное развитие благодаря своей эффективности.

3.3 Объединение глубокой сверточной нейронной сети с рекуррентной нейронной сетью

Основой LSTM-сети является ячейка памяти, кодирующая знания о наблюдаемых входных данных в каждый момент времени. Поведение ячейки

управляется затворами, в роли которых используются слои, применяемые мультипликативно. Используется три затвора (рисунок 3.8):

- забыть текущее значение ячейки,
- прочитать входное значение ячейки,
- вывести новое значение ячейки.

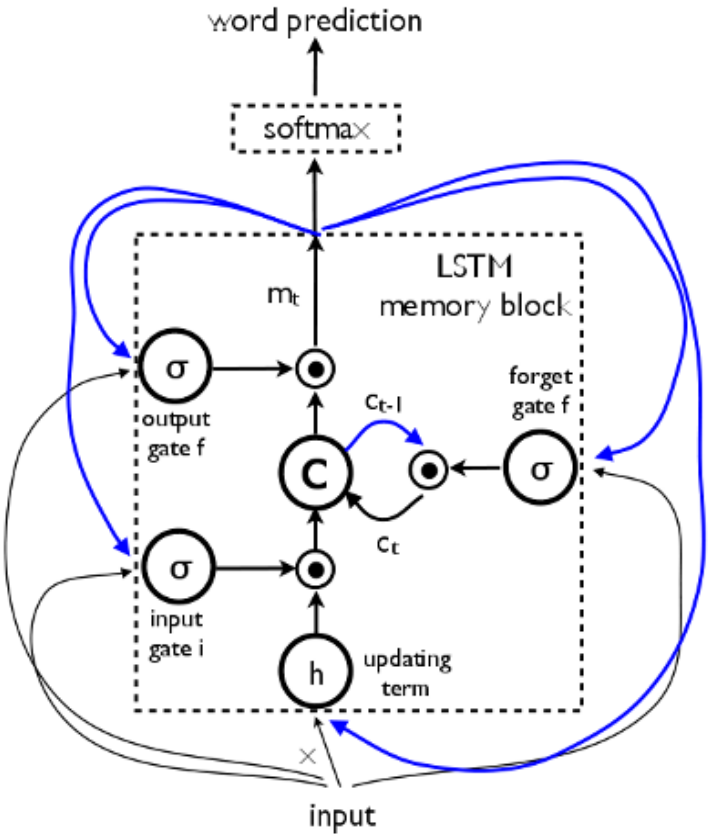


Рисунок 3.8 Рекуррентная нейросеть с длительной краткосрочной памятью

Блок памяти содержит ячейку «c», управляемую с помощью трёх затворов. Синим показаны рекуррентные подключения: вывод «m» в момент времени (t–1) подаётся обратно в память во время «t» через три затвора; значение ячейки подаётся обратно через затвор «забыть»; предсказанное слово в момент времени (t–1) подаётся обратно в дополнение к выходу памяти «m» в момент времени «t» в Softmax для предсказания тега.

В качестве входа используются функции сверточной нейронной сети глубокого обучения (полученные с помощью GoogLeNet). Затем выполняется обучение модели с LSTM на основе сочетаний этих функций сверточной

нейронной сети глубокого обучения и меток для заранее обработанных изображений.

Копия памяти LSTM создаётся для каждого изображения и для каждой метки таким образом, что все LSTM используют одинаковые параметры. Вывод (m)×(t−) LSTM в момент времени (t –) подаётся в LSTM в момент времени (t).

3.4 Реализация алгоритма распознавания эмоций речи на основе LSTM-сетей глубокого обучения

Сегодня существует больше десятка библиотек глубокого обучения, некоторые из которых обладают узкой спецификой. Основываясь на рекомендательной информации из широкого круга источников, для дальнейшего рассмотрения было выбрано три библиотеки: Torch, Theano и Caffe.

Библиотека Torch разработана как библиотека для вычислений в научных целях и поддерживает множество технологий. Библиотека позволяет гибко работать с нейросетями на достаточно низком уровне. Для реализации нейросети в Torch необходимо написать собственный цикл обучения, в котором объявляется функция замыкания, вычисляющая ответ сети. Это замыкание передаётся в функцию градиентного спуска для обновления весов сети. Использование Torch не создаёт серьёзных затрат по времени написания программного кода, кроме того, сети Torch обладают превосходящей сети других фреймворков скоростью классификации и самой обширной документацией.

Theano существует в первую очередь как расширение языка Python, позволяющее эффективно вычислять математические выражения, содержащие многомерные массивы. В библиотеке реализован базовый набор инструментов для построения искуственных нейронных сетей. Процесс создания модели и определения её параметров требует написания объёмного кода, включающего реализацию класса модели, самостоятельного определения её параметров, реализацию методов, определяющих функцию ошибки, правило вычисления

градиентов, способ изменения весов. Theano является самым гибким фреймворком для построения искусственных нейронных сетей.

Caffe [29] реализована на C++ и имеет обёртки для Python и Matlab. Топология нейронной сети, исходные данные и способ обучения задаются с помощью конфигурационных protobuf-файлов (технология и протокол сериализации данных, превосходящий по эффективности xml и json) в формате prototxt. Построение структуры сети выполняется с простотой, удобством и наглядностью. Из рассматриваемых библиотек является самой удобной в использовании, кроме того, превосходит другие рассматриваемые фреймворки в скорости обучения. Библиотека Caffe поддерживается достаточно большим сообществом разработчиков и пользователей и на сегодняшний день является самой распространённой библиотекой глубокого обучения широкого круга применения.

3.4.1 Выбор архитектуры нейронной сети

Для реализации нейронной сети была выбрана библиотека Caffe ввиду её удобной и прозрачной структуры, высокой скорости разработки при её использовании, достаточно полной документации и возможности обращения к широкому сообществу разработчиков, знакомых с ней. Кроме того, Caffe демонстрирует отличные результаты тестирования производительности. Caffe поддерживает CUDA, и сборка библиотеки с имплементацией кода для Nvidia GPU даёт колоссальный прирост в скорости работы нейросети, поскольку многие вычислительные операции в процессе функционирования нейросетевых классификаторов реализуются в несколько раз или на порядок быстрее на графических чипах.

Модель CAFFE.

Элементарной логической единицей процесса в Caffe является blob. На уровне представления blob – это N-мерный массив, хранящийся в памяти в C совместимом стиле. Blob предоставляет унифицированный интерфейс для

хранения любых данных, необходимых нейросети: входных данных, параметров модели,инструментов оптимизации.

Обычный blob для хранения входных изображений – это четырёхмерный массив, состоящий из номера N, канала K, высоты кадра H и ширины кадра W. Таким образом, в четырёхмерном blob, значение с индексом (n, k, h, w) физически располагается по индексу ((n * K + k)* H + h) * W + w. Канал K – это K-мерное пространство признаков, то есть для пикселя RGB-изображения K = 3.

3.4.2 Реализация алгоритмов и обучение нейросетей

3.4.2.1 Использованное оборудование и программное обеспечение

Разработка всех основных компонентов системы,реализация классификаторов, обучение, тестирование и валидация классификаторов производилась на высокопроизводительном компьютере под управлением операционной системы Ubuntu 14.04.LTS. Характеристики компьютера приведены в таблице 3.1.

Таблица 3.1 Характеристики используемого для вычислений ПК

Материнская плата	ASUS Z170-A
Центральный процессор	Intel Core i5-7500(4/4)
Видеоадаптер	2x Palit GeForce GTX 1060 3Gb GDDR5
Оперативная память	GeIL 2x16Gb DDR4 2133MHz
Жесткий диск	SSD Samsung 850 Evo 500Gb
Программное обеспечение	Ubuntu 14.04.LTS + caffe + DIGITS

Из особенностей вычислительного процесса при использовании нейросетевых классификаторов, можно выделить требования к скорости многократного выполнения двухмерной свёртки, скорости матричных вычислений и гибкости параллельных вычислений. Поэтому явным преимуществом при реализации системы будет использование графического

процессора, в частности поддерживающего технологию CUDA. Для обучения, тестирования и проверки использовалось два графических видеоадаптера на базе Nvidia GPU GeForce GTX 1060 мощностью 3.9 терафлопс каждый, таким образом общая приблизительно равна 7.5 терафлопс. Подробные технические характеристики видеоадаптера представлены в таблице 3.2.

Таблица 3.2 Технические характеристики GPU

Графический процессор	GeForce GTX 1060
Базовая частота видеоадаптера	1506MHz
Объем видеопамяти	3Gb
Тип памяти	GDDR5
Эффективная частота памяти	8GHz
Разрядность шины памяти	192 бит

3.4.2.2 Выбор и подготовка данных для проведения эксперимента

Для обучения нейронной сети была использована база данных IEMOCAP Южно-Калифорнийского университета. База данных представляет собой 12 часов аудио-видео данных(диалогов) 10 дикторов(5 мужчин и 5 женщин), с техническими данными(транскрипции диалогов, оценки эмоций, координаты захвата движений и т. д.).

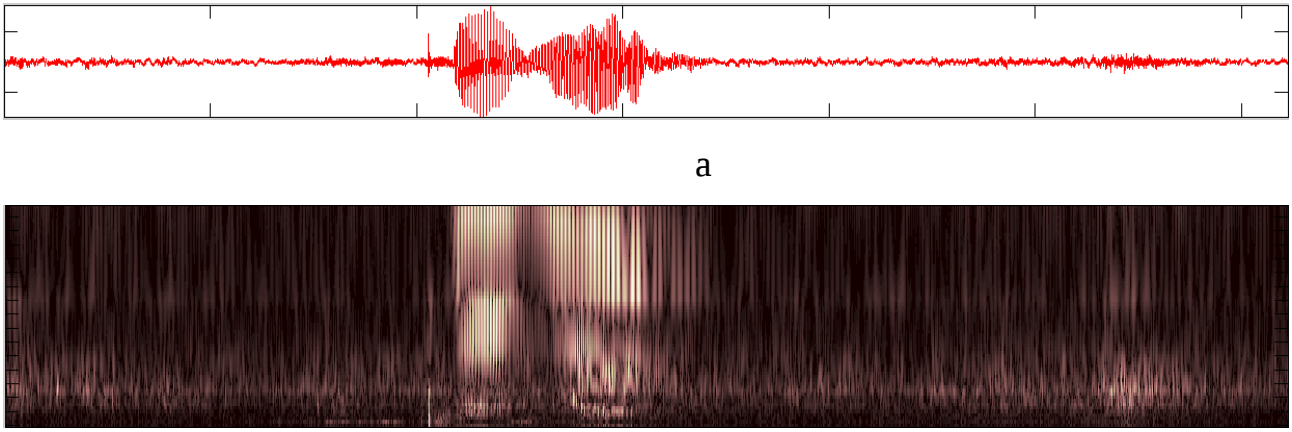
Каждое высказывание было преобразовано с помощью вейвлета Морле.

Вейвлеты – это семейства функций, локальных по времени и по частоте, в которых все функции получаются из одной посредством её сдвигов и растяжений по оси времени. Существует набор классических функций, используемых в вейвлет-анализе. К ним относятся вейвлет Хаара, вейвлет Морле, вейвлет MexicanHat, вейвлет Добеши. На практике вейвлет-анализом называется поиск произвольного паттерна на изображении при помощи свёртки изображения с моделью этого паттерна. Классические вейвлеты используются для сжатия или классификации изображений.

Преимущества проведения непрерывного преобразования с вейвлетом

Морле: 1) алгоритм реализации достаточно прост 2) ясный физический смысл решения дифференциальных уравнений в частных производных, относящихся к диффузионному типу, позволяет выявить характерные свойства вейвлет-образов функций.

На рисунке 3.9 приведён пример преобразования непрерывного непериодического сигнала.



а

б

Рисунок 3.9 Непрерывное преобразование Морле (а — исходный сигнал, б — преобразованный сигнал)

Преобразование производилось при помощи самостоятельно разработанной программной утилиты «cwt», написанной с использованием языка программирования C++. Листинг программы приведён в приложении. Также была использована дополнительная библиотека «k52»[30].

Для исключения влияния физического размера исходных данных на обучение нейронной сети, вычисление вейвлета производилось для фрагментов данных длиной 0.5 секунды. Полученный вейвлет сохранялся в виде монохромного изображения формата PNG размером 1600*512 пикселей.

Описание эмоций были непосредственно взяты из набора данных IEMOCAP — их оценка была произведена сообществом студентов Южно-Калифорнийского университета.

Таким образом для обучения нейронной сети были подготовлены

обработанные вейвлетом Морле высказывания представленные в виде изображений (рисунок 3.10) и текстовые файлы, содержащие описание эмоций каждого высказывания.

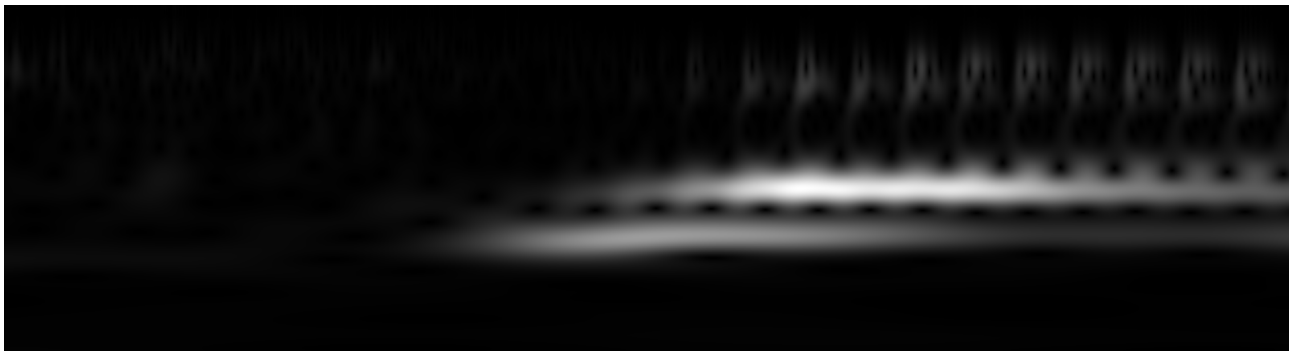


Рисунок 3.10 Пример входных данных для нейронной сети

3.4.3 Результаты испытания системы

Проверочная выборка формировались исходя из 10% от обучающей выборки (рисунок 3.11).

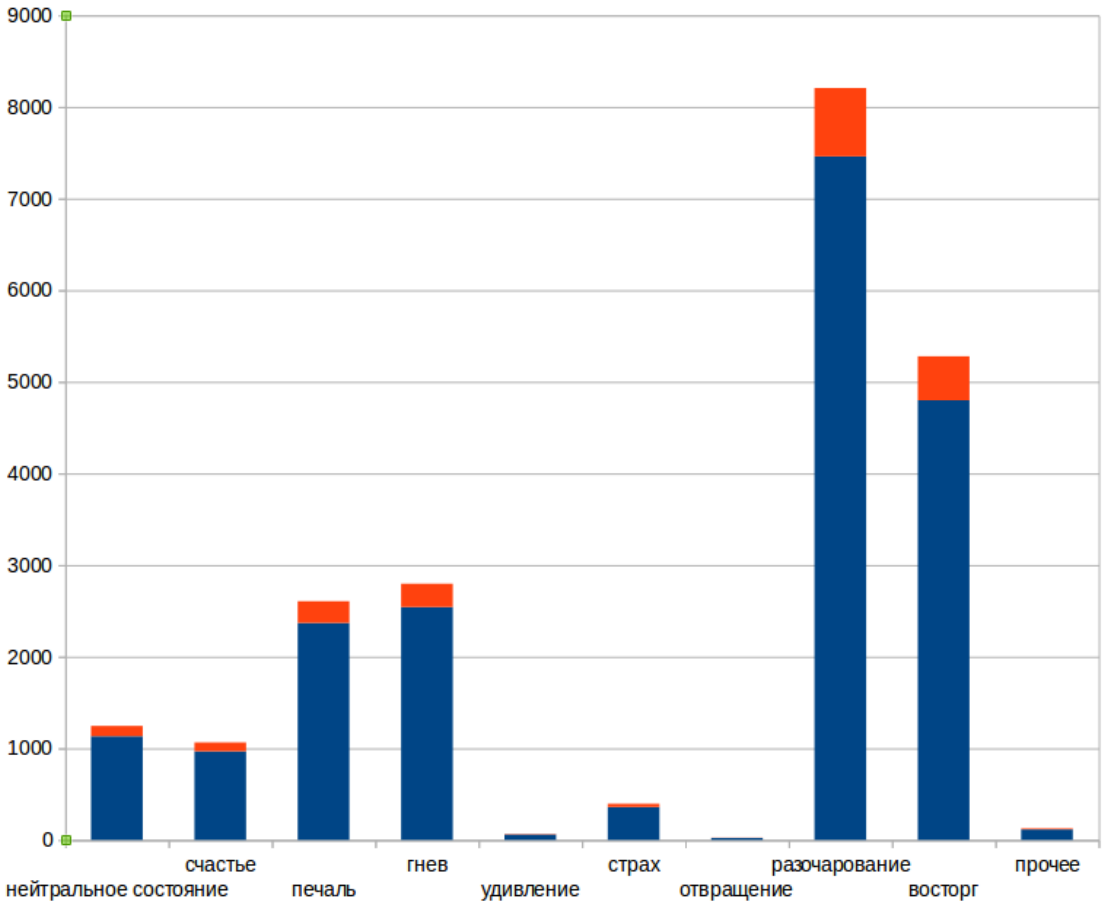


Рисунок 3.11 Соотношение количества эмоций (красным — проверочная выборка, синим — обучающая выборка)

Процесс обучения занял более 56 часов. График демонстрирующий результаты в процессе обучения приведён на рисунке 3.12. Результаты работы классификатора приведены в таблице 3.3.

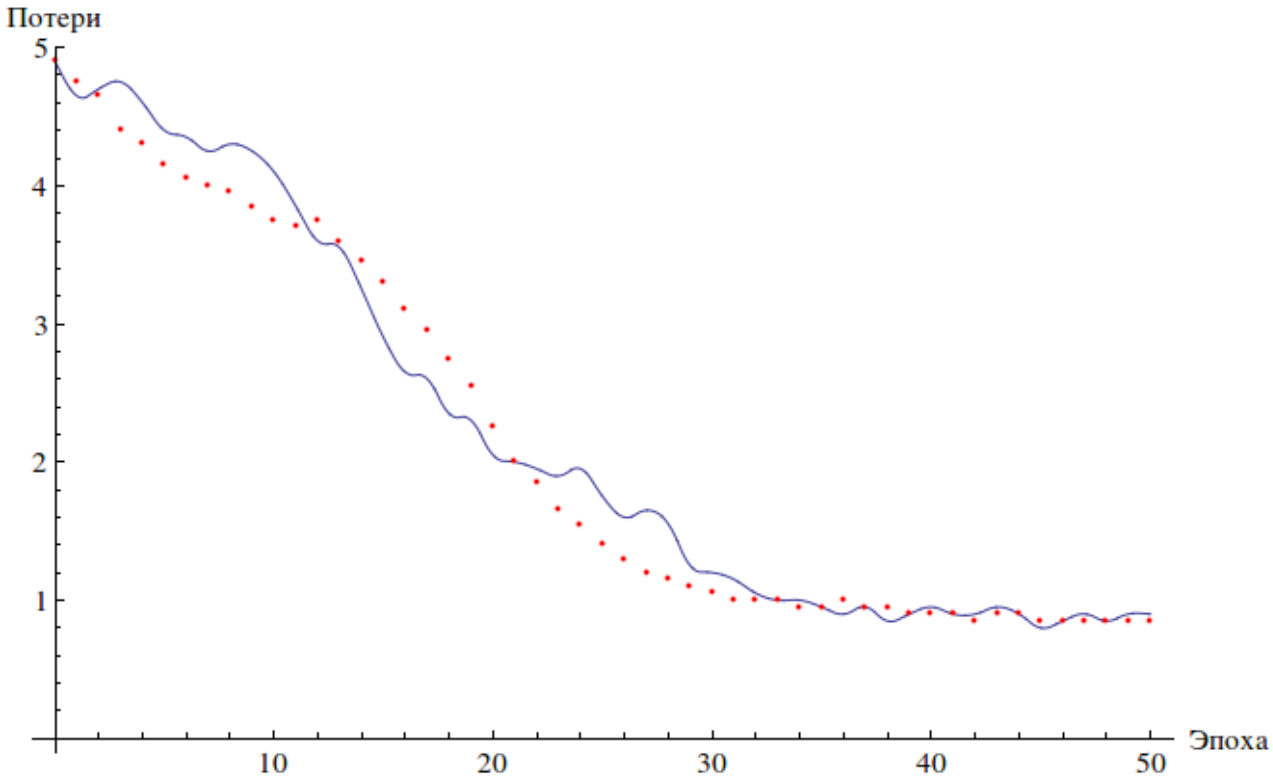


Рисунок 3.12 График функции потерь на обучающей(линия) и проверочной(точки) выборках

Точность классификатора составила 49% для 10 эмоций на проверочной выборке (рисунок 3.13). Данный показатель связан, в первую очередь, с достаточно качественной, однако неоднородной обучающей выборкой (рисунок 3.11). На каждую эмоцию приходится неодинаковое количество обучающих данных, таким образом чётко прослеживается неоднородность обучения классификатора. Ещё одна трудность связана со временными свойствами эмоций. Часто почти все высказывание не имеет эмоций(говорящий находится в нейтральном состоянии), а эмоциональность содержится только в нескольких словах или фонемах в высказывании. Проблемы распознавания эмоций может быть переформулирована в математических терминах как задача классификации.

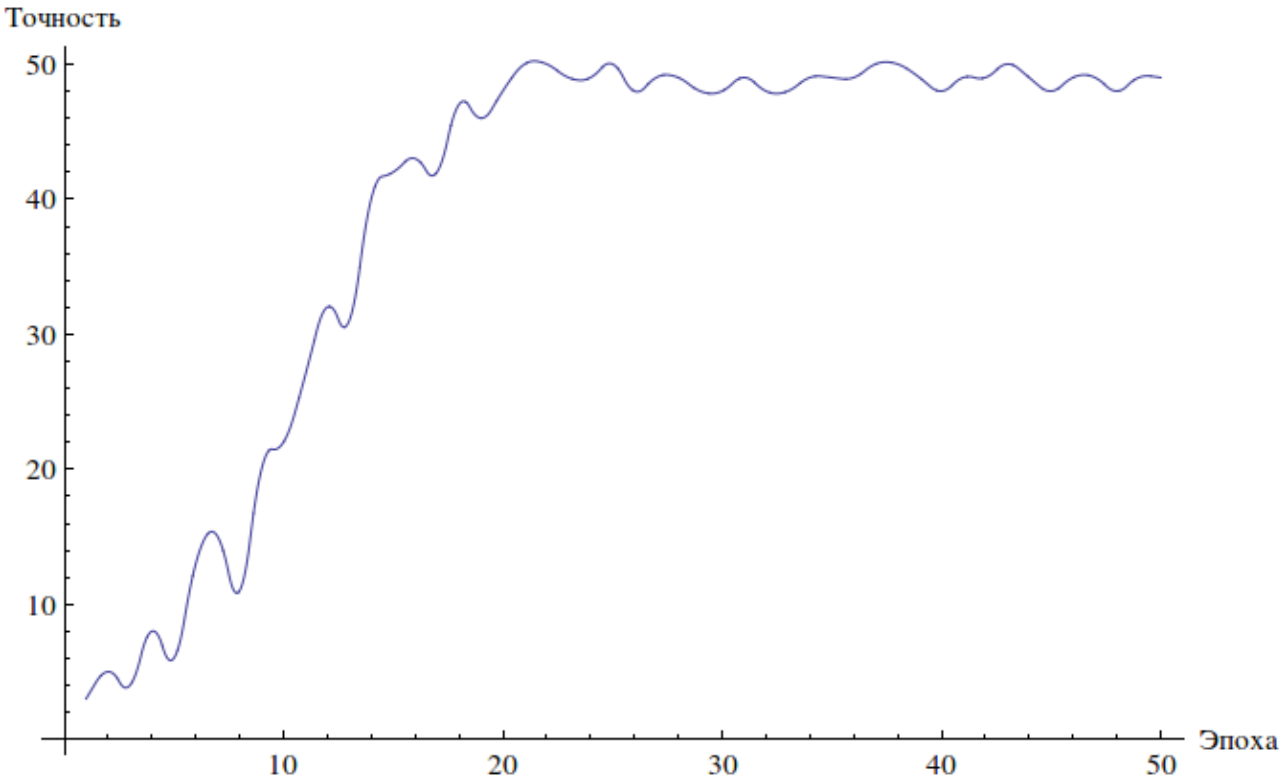


Рисунок 3.13 График функции точности на проверочной выборке.

Таблица 3.3 Матрица спутывания

Эмоция \ Вероятность	Нейтральное состояние	Счастье	Печаль	Гнев	Удивление	Страх	Отвращение	Разочарование	Восторг	Прочее
Нейтральное состояние	**66**	4	5	5	1	1	0	14	4	0
Счастье	4	**69**	3	2	2	1	1	6	11	1
Печаль	2	1	**73**	1	3	0	1	19	0	0
Гнев	4	5	2	**77**	1	1	0	6	3	1
Удивление	2	3	1	0	**11**	0	1	43	38	1
Страх	7	9	11	4	3	**16**	2	25	22	1
Отвращение	8	6	9	2	3	4	**13**	41	12	2
Разочарование	1	0	5	1	3	2	4	**83**	1	0
Восторг	1	11	0	0	6	0	0	0	**82**	0
Прочее	3	8	3	12	3	1	1	39	26	**4**
Итого: 49,4										

В проведённом в 2014 году совместном исследовании [31] компании Майкрософт и факультета Компьютерных наук и инжиниринга университета Огайо с использованием сверточных нейронных сетей для первичной обработки исходных данных (также обучающая выборка на основе IEMOCAP) использовались мел-частотные кепстральные коэффициенты. Итоговая точность в среднем составила 47% (рисунок 3.14).

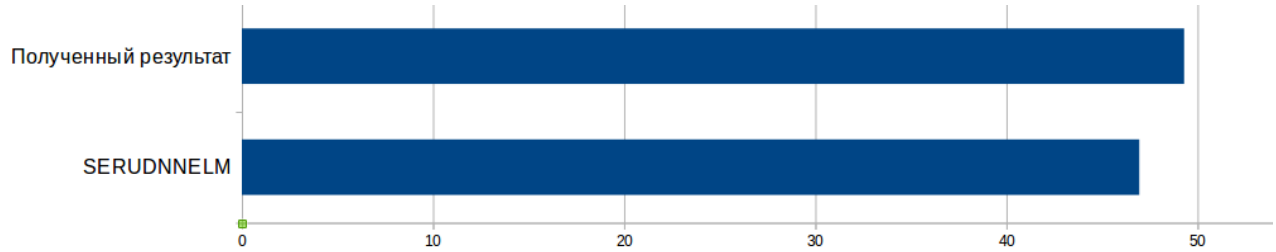


Рисунок 3.14 Сравнение результатов алгоритма SERUDNNELM и результат полученного в результате работы обученного алгоритма

ЗАКЛЮЧЕНИЕ

В ходе дипломной работы выполнены следующие задачи:

- изучены основные принципы и существующие методы решения задач машинного обучения, распознавания образов и компьютерного зрения,
- изучены технологии построения нейронных сетей глубокого обучения на базе графических видеоадаптеров.
- проведён анализ и выбор необходимых программных пакетов.
- разработан алгоритм распознавания эмоций речи с использованием LSTM-сетей глубокого обучения.
- произведены необходимые преобразования с исходными данными для подготовки их к обучению.
- обучен LSTM-классификатор с использованием программного пакета CAFFE и получен результат 49% точности для 10 эмоций, что сравнимо с результатами предыдущих исследований в данном направлении.

Проведённые исследования позволили сделать вывод, что улучшить данный алгоритм можно путём более точной подготовки исходных данных для обучения:

- качественно — как было упомянуто выше, зачастую почти все высказывание не имеет эмоций, а эмоциональность содержится только в нескольких словах или фонемах в высказывании, что хорошо прослеживается по результатам эксперимента. Таким образом необходимо либо давать для обучения только участки проявления эмоции, либо усовершенствовать нейронную сеть и передавать на вход не только исходные данные, но и информацию о положении эмоции в высказывании, а высказывания в диалоге.
- количественно — для более качественного обучения нейронной сети желательно использовать обучающую выборку, в которой количество данных на класс (эмоцию) должно быть приблизительно одинаковым.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition / Rabiner L. // IEEE Press. – 1988. – P. 257 - 286.

2. Young, S. The HTK Book (for HTK v. 3.4) / Young S., Evermann G., Gales M., Hain T., Kershaw D., Liu X., Moore G., Odell J. Ollason D., Valtchev V., Woodland P. - Cambridge

3. Multi-style training for robust isolated-word speech recognition / R. P. Lippmann, E. A. Martin, and D. B. Paul // in Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing, Dallas, TX. – 1987. – P. 705–708.

4. A Comparative Study of Traditional and Newly Proposed Features for Recognition of Speech Under Stress / Bou-Ghazale S.E., Hansen J.H.L. // Speech and Audio Processing. – 2000. - № 8. - P. 429 – 442.

5. Using neutralized formant frequencies to improve emotional speech recognition / Davood Gharavian, Mansour Sheikhan, Farhad Ashoftedel // IEICE Electronics Express. – 2011. - № 14. - P. 1155 – 1160.

6. Speech Under Stress: Analysis, Modeling and Recognition / Hansen J. H. L., Sanjay Patil // Lecture Notes in Computer Science. – 2007. - № 4343. - P. 108 – 137.

7. Towards Robust Spontaneous Speech Recognition with Emotional Speech Adapted Acoustic Models / Vlasenko B., Prylipko D.,Wendemuth A. // 35th German Conference on Artificial Intelligence (KI-2012). – 2012. – P. 103 - 107.

8. A study on speaker adaptation of the parameters of continuous density hidden Markov models / C.-H. Lee, C.-H. Lin, B.-H. Juang // IEEE Trans. Signal Processing. – 1991. - № 39. – P. 806–814.

9. Emotions and speech: Some acoustical correlates / Williams C., Stevens K. // J. Acoust. Soc. Amer. – 1972. – № 52. – P. 1238 – 1250.

10. Информация об эмоциональных состояниях в речевой интонации / Витт Н.В. // Вопросы психологии. – 1965. – № 3. – P. 89 – 103.

11. Динамика формант в спектре слышимой речи как объективный показатель эмоций / Тищенко А.Г. // Космическая биология и медицина. – 1968. – № 5. – P. 82 – 86.

12. Лукьянов А.Н. Сигналы состояния человека–оператора / Лукьянов А.Н., Фролов М.В. – М., «Наука», 1969. – 246 р.

13. Характеристики речевого сигнала как индикатор эмоционального состояния оператора / Галунов В.И., Манеров В.Х. // Материалы 4–ой Всесоюзной конференции по инженерной психологии и эргономике. – Ярославль, 1974. – P. 3 – 8.

14. Vocal communication of emotion / Johnstone T., Scherer K.R. // В кн.: «Речь и эмоции»/ In: Lewis, M., Haviland, J. (Eds.), Handbook of emotion, second ed. Guilford, New York. – P. 220 – 235.

15. A cross–cultural investigation of emotion inferences from voice and speech: Implications for speech technology / Scherer K. R. // В кн.: «Речь и эмоции»/ In: Proc. Interspeech. – Beijing, China, 2000. – P. 379 – 382.

16. Emotion recognition in human–computer interaction / Cowie R., Douglas–Cowie E., Tsapatsoulis N., Kollias S., Fellenz W., Taylor, J. // IEEE Signal Process. – 2001. – № 18. – P. 32 – 80.

17. Mitchell T., Machine Learning — McGraw Hill, 1997 — 414 с.

18. Флах П. Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных. – М.: ДМК Пресс, 2015. – 400 с.

19. Горелик А.Л., Скрипкин В.А. Методы распознавания. М., Высшая школа, 1977.

20. Шапиро Л., Стокман Дж. Компьютерное зрение — М.:Бином. Лаборатория знаний, 2006 — 752 с.

21. Уоссермен Ф. Нейрокомпьютерная техника: Теория и практика. – М.: Мир,1992. – 184 с.

22. Хайкин С. Нейронные сети: полный курс, 2-е издание. – М.:

Вильямс, 2008. – 1103 с.

23. Fukushima K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position // Biological cybernetics. – 1980. – Vol. 36, no. 4. – Pp. 193-202.

24. Gradient-based learning applied to document recognition / Y. LeCun [et al.] // Proceedings of the IEEE. – 1998. – Vol. 86, no. 11. – Pp. 2278-2324.

25. Toshev A., Szegedy C. Deeppose: Human pose estimation via deep neural networks // Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on. – IEEE. 2014. – Pp. 1653-1660.

26. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups / G. Hinton [et al.] // Signal Processing Magazine, IEEE. – 2012. – Vol. 29, no. 6. – Pp. 82-97.

27. Large-scale video classification with convolutional neural networks / A. Karpathy [et al.] // Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on. – IEEE. 2014. – Pp. 1725-1732.

28. Koltsova O., Shcherbak A. ‘LiveJournal Libra!’: The political blogosphere and voting preferences in Russia in 2011–2012 // New Media & Society. – 2014.

29. Caffe | Deep Learning Framework. URL: Режим доступа: http://caffe.berkeleyvision.org/ Дата доступа — 12.04.2017

30. k52 is a set of C++ libraries aimed to facilitate scientific experiments in the fields of signal processing and sound analysis with the strong incline into OOP, flexibility and readability. URL: Режим доступа: https://github.com/PavelKovalets/k52 Дата доступа — 24.04.2017

31. Speech Emotion Recognition Using Deep Neural Network and Extreme Learning Machine. URL: Режим доступа: https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/IS140441.pdf Дата доступа — 06.02.2017

Действующий личный сайт

https://kuntsevichsergey.github.io

ПРИЛОЖЕНИЕ

```
1  #include <iostream>
2  #include <memory>
3  #include <boost/shared_ptr.hpp>
4  #include <fftw3.h>
5  #include <png++/image.hpp>
6  #include <png++/gray_pixel.hpp>
7  #include <k52/dsp/wavelet.h>
8  #include <k52/dsp/transform/wavelet/linear_scale.h>
9  #include <k52/dsp/transform/wavelet/logarithmic_scale.h>
10 #include <k52/dsp/transform/wavelet/fast_wavelet_transform.h>
11 #include <io/sound_file.h>
12
13 typedef png::image<png::gray_pixel> PngImage;
14
15 void usage()
16 {
17    std::cout << "Usage: cwt <input_filename> <emotion>" << std::endl;
18 }
19
20 void save_png(const std::string& filename,
21            std::vector<std::vector<double>>& wavelet)
22 {
23
24    PngImage image(wavelet[0].size(), wavelet.size());
25    double max = std::numeric_limits<double>::min();
26    double min = std::numeric_limits<double>::max();
27    for (int i = 0; i < wavelet.size(); ++i)
28    {
29       max = std::max(max, *std::max_element(wavelet[i].begin(), wavelet[i].end()));
30       min = std::min(min, *std::min_element(wavelet[i].begin(), wavelet[i].end()));
31    }
32
33    for (int row_index = 0; row_index < wavelet.size(); ++row_index)
34    {
35       PngImage::row_type& row = image.get_row(row_index);
36       for (int column_index = 0; column_index < wavelet[0].size(); ++column_index)
37       {           double value = 255 * (wavelet[row_index][column_index] - min) / (max - min);
38          row[column_index] = value;
39       }
40    }
41
42    image.write(filename);
43 }
44
45 int main(int argc, char *argv[])
46 {
```

```
47    if (argc != 3)
48    {
49      usage();
50      return 1;
51    }
52
53    fftw_import_wisdom_from_filename("fftw.wisdom");
54    std::string sound_file_name = std::string(argv[1]);
55    std::string output_file_name = std::string(argv[2]);
56
57    io::SoundFile<double> sound_file(sound_file_name);
58    k52::dsp::Wavelet::ScaleType type = k52::dsp::Wavelet::Log;
59
60    std::vector<std::complex<double> > data;
61    std::transform(sound_file.data().begin(), sound_file.data().end(),
62              std::back_inserter(data),
63              [](double real) { return std::complex<double>(real, 0); } );
64
65    k52::dsp::IScale::shared_ptr scale(new k52::dsp::LogarithmicScale(0.5, 8.5, 512));
66    k52::dsp::FastWaveletTransform wt(scale, data.size());
67    k52::dsp::IWavelet::shared_ptr w(new k52::dsp::MorletWaveletFunction());
68
69    std::vector<std::vector<std::complex<double> > > complex_result = wt.Transform(data, w);
70
71    std::vector<std::vector<double> > wavelet;
72    std::transform(complex_result.begin(), complex_result.end(), std::back_inserter(wavelet),
73              [](const std::vector<std::complex<double> >& frame) -> std::vector<double> {
74                std::vector<double> power;
75                std::transform(frame.begin(), frame.end(), std::back_inserter(power),
76                          [](std::complex<double> value)
77                           { return sqrt(powf(value.real(), 2) + powf(value.imag(), 2)); }
78                          );
79                return power;
80              });
81
82    fftw_export_wisdom_to_filename("fftw.wisdom");
83    save_png(output_file_name, wavelet);
84
85    return 0;
86  }
87
```