Министерство науки и высшего образования Российской Федерации
Ярославский государственный университет им. П. Г. Демидова
Кафедра цифровых технологий и машинного обучения

А. И. ТОПНИКОВ

ПРИМЕНЕНИЕ НЕЙРОННЫХ СЕТЕЙ В ЗАДАЧАХ ОБРАБОТКИ РЕЧЕВЫХ СИГНАЛОВ

Учебно-методическое пособие

Ярославль
ЯрГУ
2022

УДК 004.934(075.8)
ББК 387-013я73
Т58

Рекомендовано
Редакционно-издательским советом университета
в качестве учебно-методического пособия. План 2022 года.

Рецензент
кафедра цифровых технологий и машинного обучения ЯрГУ

Топников, Артем Игоревич.

Применение нейронных сетей в задачах обработки
Т58 **речевых сигналов** : учебно-методическое пособие /
А. И. Топников ; Яросл. гос. ун-т им. П. Г. Демидова. –
Ярославль : ЯрГУ, 2022. – 36 с.

Приводятся краткие сведения по применению нейронных сетей в задачах обработки, анализа, распознавания и синтеза речевых сигналов.

Пособие предназначено для студентов, обучающихся по дисциплине «Цифровая обработка речевых сигналов».

Материал может быть использован также при подготовке студентами курсовых и выпускных квалификационных работ.

УДК 004.934(075.8)
ББК 387-013я73

Введение

Первые задокументированные шаги, направленные на реализацию и применение искусственных нейронных сетей, относятся к концу 1950-х годов. Но наработки, полученные на ранних этапах развития теории нейронных сетей, не привели в большинстве случаев к коммерциализуемым практическим результатам. Потребовалось несколько десятилетий, на которые пришлось изобретение алгоритма обратного распространения ошибки, совершенствование градиентных методов оптимизации, появление различных архитектур нейросетей и специализированных процессоров для их обучения, прежде чем нейросетевые алгоритмы смогли вырваться в лидеры при решении широкого круга задач, включающего в себя в том числе и задачи обработки и анализа речевых сигналов.

Автор данного пособия не ставит целью проследить всю историю развития теории, которая привела к появлению современных нейросетевых алгоритмов обработки речевых сигналов, равно как и не ставит целью отобразить современный технологический ландшафт данной предметной области. Выбранный формат позволяет познакомить читателя лишь с базовыми принципами построения и использования нейросетевых алгоритмов, а также с некоторыми перспективными направлениями исследований в данной области.

Первый раздел пособия посвящен признакам речевых сигналов, наиболее часто используемым в современных нейросетевых алгоритмах, а также ряду смежных вопросов. Второй раздел – задаче подавления шума в речевых сигналах и вопросам оценки их качества. В третьем – описываются основные задачи распознавания и синтеза речевых сигналов. В этом разделе, кроме задачи преобразования речи в текст и распознавания отдельных слов, также описываются задачи, обобщенно называемые распознаванием диктора. Каждый из трех разделов завершается блоком практических заданий для самостоятельного выполнения.

Контрольные вопросы, приведенные в конце издания, позволяют проверить качество усвоения теоретических знаний, полученных при изучении пособия, а приведенный список источников позволяет перейти к более глубокому изучению отдельных направлений исследований и конкретных алгоритмов.

1. Признаки речевых сигналов

Многие книги по теории и применению нейронных сетей начинаются с описания простейших нейронных сетей. Очень часто в качестве примера используются алгоритмы распознавания изображений, в которых вмешательство в исходный сигнал, подаваемый на вход сети, минимально. Однако в области речевой обработки, как и одно-два десятилетия назад, большинство алгоритмов работает с сигналами, представленными в частотной области или иной области трансформант. Зачастую вопрос выбора формы представления сигнала для решения конкретной задачи является не менее важным, чем вопросы, связанные с последующей обработкой. Этап формирования признаков (представления) сигнала необходим для создания компактного и информативного представления данных, с которыми работает конкретный алгоритм. В рамках концепции глубокого обучения роль выбора признаков может показаться менее значительной, однако знание и понимание базовых основ в этой области все же необходимы.

В последние десятилетия в обработке речевых и других звуковых сигналов вместо традиционных спектрограмм, полученных на основе оконного преобразования Фурье (рис. 1), часто прибегают к мел-частотным кепстральным коэффициентам (МЧКК) и спектрограммам на их основе. Одним из показателей широкой распространённости и популярности этого типа признаков служит то, что англоязычная аббревиатура MFCC (Mel Frequency Cepstral Coefficients) настолько укрепилась в профессиональном лексиконе, что зачастую используется без предварительной расшифровки.

В чем особенность этих коэффициентов и как они вычисляются? Можно постараться найти ответ в самом название признаков. Самым простым и не требующим пояснения является слово «коэффициенты». Разберемся, что такое кепстр и почему коэффициенты не просто частотные, а мел-частотные.

Начнем с кепстра. Понятие кепстр впервые возникло в 1962 году и связано с исследованиями в области радиотехники. Группа ученых установила возможность оценки задержки отраженного радиосигнала на основе вычисления преобразования Фурье от логарифма спектра мощности. Эти параметры не являются спектром в традиционном понимании, поэтому создатели метода предложили в качестве названия слово, в котором часть букв слова «спектр» на английском языке записана в обратном порядке. Аналогичным образом ими были трансформированы

слова «частота» и «фаза». В настоящее время кестральный анализ рассматривается в качестве вида гомоморфной обработки и находит применение в самых разных областях.

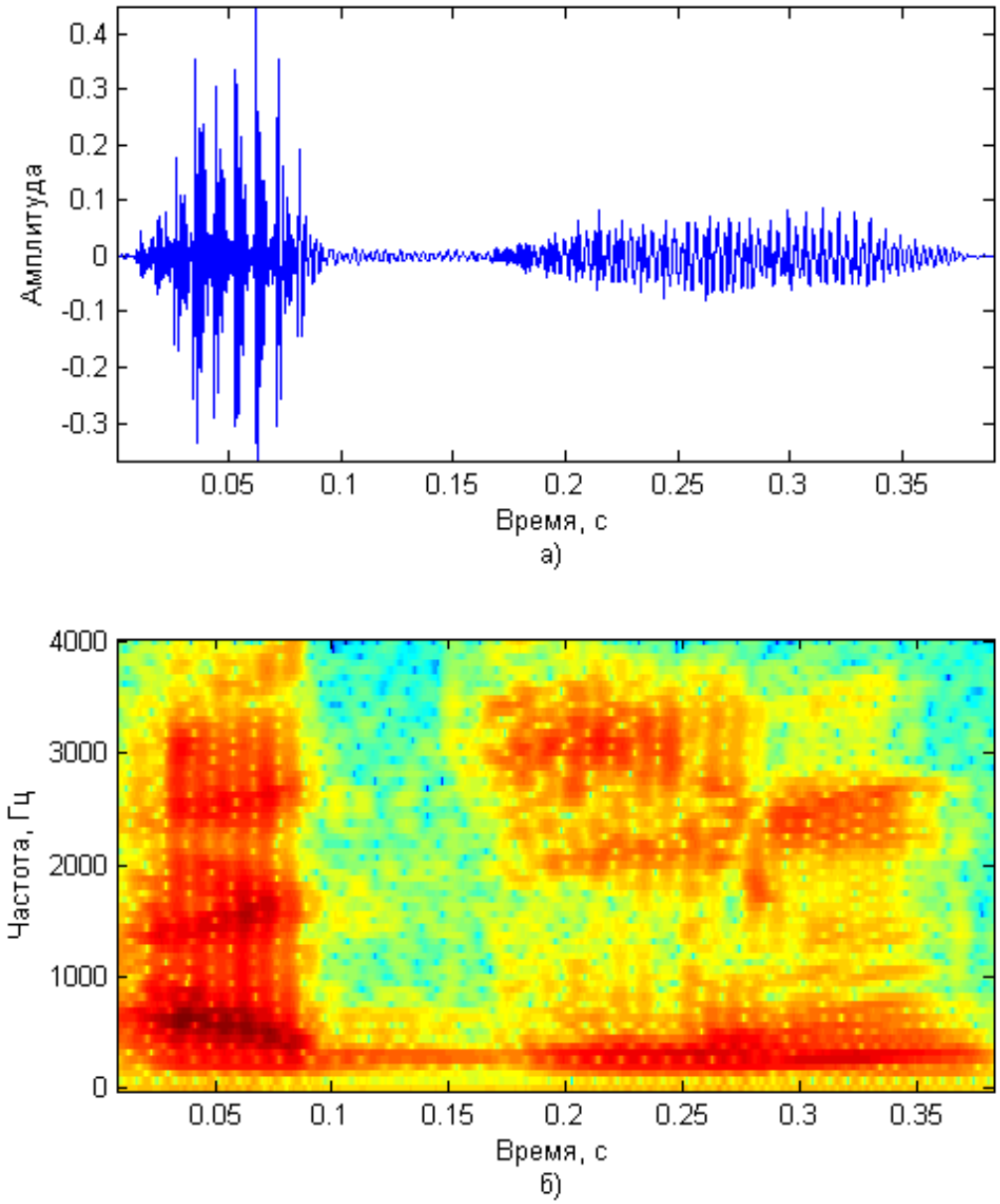

Рис. 1. Представление речевого сигнала во временной (а) и частотной (б) областях

Теперь, когда мы имеем общие представления о кепстре, перейдем к рассмотрению мел-шкалы. Ее появление связано с развитием психоакустики – раздела психофизики, изучающего восприятие звука человеком. Было установлено, что высота отдельных звуков нелинейно связана с их частотой. Отдельные научные группы изучали эту

взаимосвязь, поэтому существует несколько единиц высоты звука. В настоящее время наиболее востребованными и широко используемыми являются мел-единицы. По определению звуковые колебания частотой 1000 Гц при уровне громкости 40 фон, воздействующие на человека с нормальным слухом, вызывают у него восприятие высоты звука, приравненной к 1000 мел.

Графическое изображение взаимосвязи высоты звука, выраженной в мелах, и частоты в Герцах представлено на рисунке 2. Также рассмотрим и формулы, преобразующие мелы в Герцы и в обратном направлении:

$$m = 2595 \log_{10}\left(1 + \frac{f}{700}\right) = 1127 \ln\left(1 + \frac{f}{700}\right),$$
$$f = 700\left(10^{\frac{m}{2595}} - 1\right) = 700\left(e^{\frac{m}{1127}} - 1\right),$$

где f – частота, выраженная в Герцах, m – высота звука в мелах. Стоит также учитывать, что воспринимаемая высота звуков также зависит от уровня громкости и ряда других факторов. Не говоря уже о том, что большинство звуков, окружающих человека, не являются моногармоническими и их восприятие связано со сложными механизмами, изучение и математические описание которых крайне сложно.

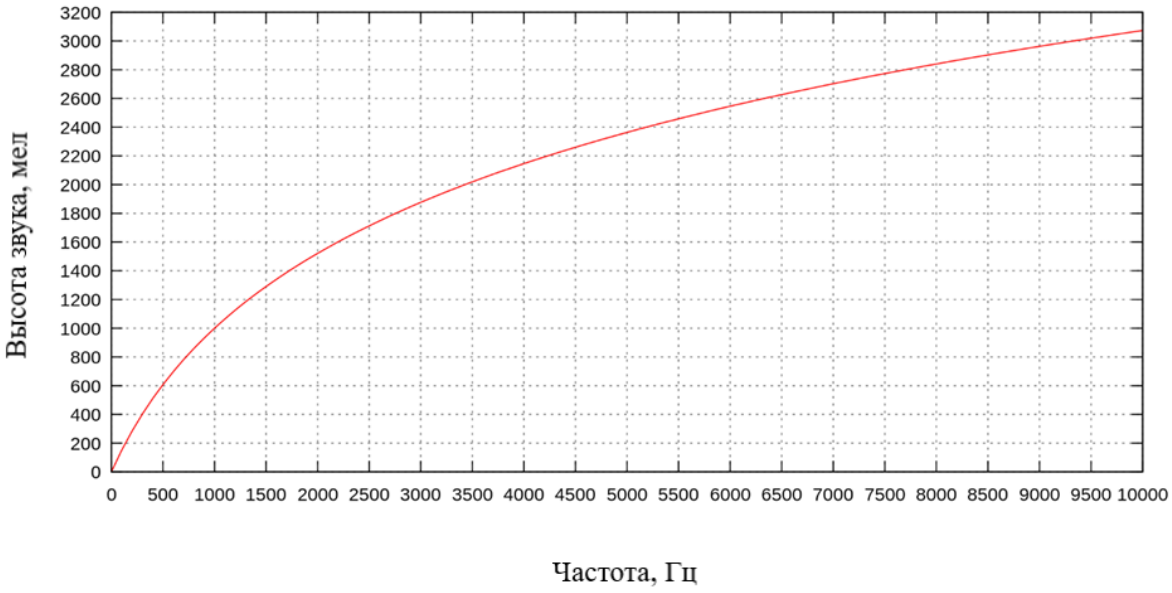

Рис. 2. Взаимосвязи высоты звука, выраженной в мелах, и его частоты в Герцах

Как было указано ранее, мел-шкала не является единственным подходом к установлению взаимосвязи между высотой звука и его частотой. Для примера рассмотрим подход, предложенный Эбехардом Цвикером. Изучая критические полосы человеческого слуха, этот исследователь предложил в 1961 году единицу высоты Барк, которую назвал в честь другого ученого Генриха Георга Баркгаузена. В качестве формулы, связывающей высоту звука в Барках с частотой в Герцах, приведем ту, что была приложена Эбехардом Цвикером:

$$z(f) = 13 \cdot \operatorname{arctg}\left(\frac{0,76f}{1000}\right) + 3.5 \cdot \operatorname{arctg}\left(\left(\frac{f}{7500}\right)^2\right).,$$

где *f* – частота, выраженная в Герцах, *z* – высота звука в Барках.

В настоящее время существуют несколько аппроксимаций этой зависимости, использующих в качестве основы разные функции, в том числе гиперболический синус и натуральный логарифм.

Вернемся к мел-частотным кепстральным коэффициентам. Теперь, когда все слова, составляющие это название, понятны, можно перейти к алгоритму вычисления этих признаков для произвольного звукового сигнала.

Для начала нужно вычислить амплитудный спектр фрагмента звукового сигнала. Для этого используется оконное преобразование Фурье. Длина и шаг окна, а также тип используемой оконной функции зависят от особенностей решаемой задачи. Очень часто в речевой обработке используют окна длиной от 128 до 2048 отсчетов. Стоит также напомнить, что длину временного промежутка в секундах, которому соответствует конкретная длина окна, определяет частота дискретизации. Но в любом случае выбранная длина окна определяет количество спектральных отсчетов в найденном с помощью оконного преобразования Фурье спектре. В большинстве практических случаев их число измеряется сотнями, а иногда и тысячами. Все ли эти составляющие несут достаточно много информации? Удобно ли анализировать такое представление сигнала? Для решения значительного круга задач речевой обработки ответ на этот вопрос будет отрицательным, то есть – нет. На этот ответ наталкивает не только практический опыт, накопленный за последние десятилетия специалистами в области цифровой обработки речи, но и результаты психоакустических исследований, касающихся, в частности, критических полос. Слуховой системе человека, как и многим современным

алгоритмам речевой обработки, не требуется столь детальное представление звуковых сигналов. Если в случае с человеком форма представления акустических сигналов определяется физиологией слуховой системы, то в нашем случае переход к более обобщенному, агрегированному спектральному представлению осуществляется за счет применения мел-фильтров. Банк таких фильтров можно построить и во временной области, но на практике чаще идут по пути применения треугольных функций (рис. 3), равномерно распределённых на мел-шкале к спектрограмме, полученной на предыдущем шаге. Каждый столбец исходной спектрограммы умножается на соответствующий мел-фильтр (спроецированный на частотную шкалу), после чего получается вектор чисел, по размеру равный количеству мел-фильтров. Таким образом значительно сокращается количество составляющих, а разрешение по частоте становится неравномерным – с большим акцентом на низкие частоты. Это тоже заимствование из области психоакустики.

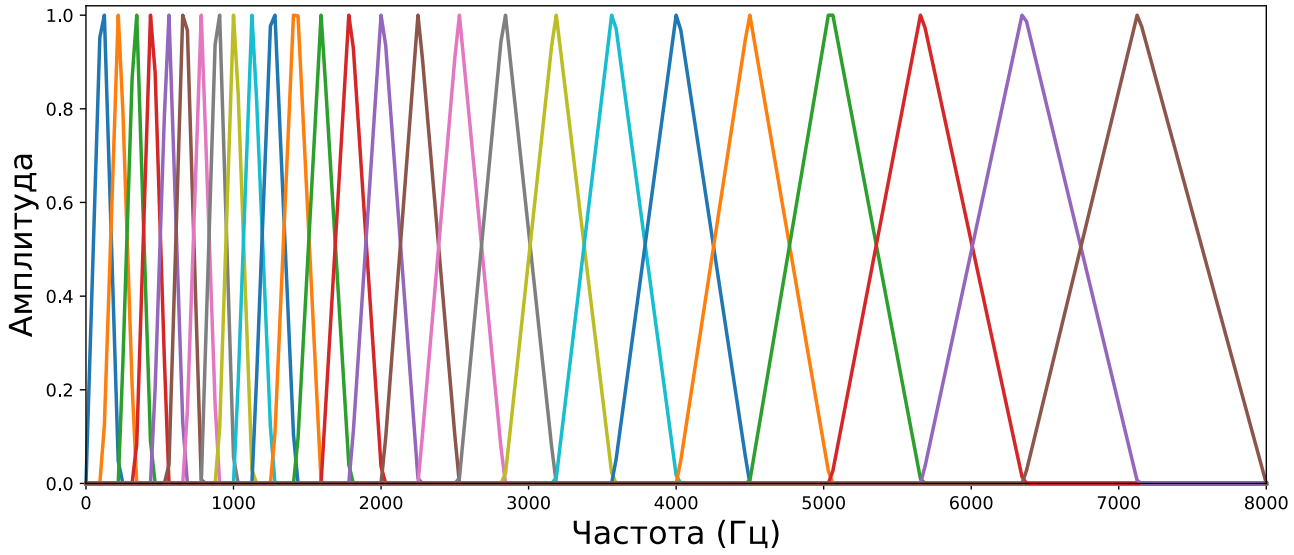

Рис. 3. Амплитудно-частотная характеристика банка треугольных мел-фильтров

То представление, которое мы получили на этом шаге, уже можно использовать для анализа и обработки речевых сигналов. Многие современные алгоритмы, в том числе и нейросетевые, используют мел-спектрограммы в качестве входных признаков звуковых сигналов. Однако это еще далеко не мел-частотные кепстральные коэффициенты. Это все еще спектр, пусть и расположенный на мел-шкале.

Сделаем небольшую остановку в процессе вычисления мел-частотных кепстральных коэффициентов и рассмотрим, как на практике

можно без лишних временных затрат построить мел-спектрограммы. Начнем с популярного языка программирования Python. В нем для построения мел-спектрограмм удобно воспользоваться функцией *melspectrogram* из библиотеки Librosa, на вход которой могут подаваться как отсчеты сигнала во временной области, так и заранее вычисленная спектрограмма. Также на вход функции подается значение частоты дискретизации, требуемая длина окна преобразования Фурье, шаг смещения окна и тип оконной функции, а также другие параметры. При визуализации мел-спектрограммы одна из осей размечена в Герцах; это не должно пугать: сами составляющие расположены согласно мел-шкале, а пересчет эквивалентных им значений в Герцах произведен для удобства анализа спектрограммы.

Мел-спектрограмму можно также вычислить и построить в среде моделирования Matlab. Для этого нужно воспользоваться функцией *melSpectrogram*. Отличия с подобной функцией из библиотеки Librosa для языка Python минимальны. Самое заметное отличие – это то, что при визуализации спектрограммы по умолчанию используется другая цветовая схема. Но визуализация нужна лишь для пользователей, поскольку нейронная сеть или любой другой алгоритм работают с числами.

А теперь вернемся к вычислению мел-частотных кепстральных коэффициентов. Для получения кепстра потребуется нелинейность и еще одно преобразование. Поэтому возводим в квадрат и логарифмируем получившиеся на предыдущем этапе коэффициенты, а затем применяем еще одно преобразование. Мы могли бы опять прибегнуть к преобразованию Фурье, но на практике в качестве заключительного преобразования чаще используют дискретное косинусное. Выполнив последовательно эти шаги, получаем мел-частотные кепстральные коэффициенты.

Можно вычислить эти коэффициенты для небольшого фрагмента сигнала (одного окна), а можно разбить большой сигнал на окна (строго следующие друг за другом или перекрывающиеся) и затем от каждого окна вычислить мел-частотные кепстральные коэффициенты и из получившихся векторов сложить МЧКК-грамму (спектрограмму на основе МЧКК). Затем вектор коэффициентов или матрицу – МЧКК-грамму подают на вход нейронной сети или другого алгоритма обработки звуковых сигналов.

Выбор параметров при вычислении мел-частотных кепстральных коэффициентов зависит от особенностей решаемой задачи. В одних

приложениях мы анализируем все нюансы сигнала и работаем на уровне отдельных фонем, в других – нас интересуют обобщенные характеристики речевых сигналов (например, тембральный окрас). Поэтому к выбору длины окна преобразования Фурье на первом этапе вычисления МЧКК и требуемому числу коэффициентов на выходе нужно подходить осмысленно, учитывая особенности решаемой задачи.

Как и в случае с мел-спетрограммами, рассмотрим вычисление мел-частотных кепстральных коэффициентов на языке Python и в среде Matlab. В Python можно воспользоваться специальной функцией *mfcc* из библиотеки Librosa. Все очень похоже на функцию *melSpectrogramm*, которую мы уже рассматривали. Но для функции *mfcc* дополнительно нужно задать требуемое количество коэффициентов и тип дискретного косинусного преобразования. Стандартный формат визуализации может показаться не очень наглядным, но, как мы уже отмечали, нейронная сеть работает с числами, а визуализация в виде изображения осуществляется лишь для человека. При необходимости визуального анализа можно выбрать более информативную цветовую схему, отличную от той, что используется по умолчанию.

Если же говорить о среде Matlab, то для нахождения мел-частотных кепстральных коэффициентов разработчик предлагает две функции: *cepstralCoefficients* и *mfcc*. Первая, как видно из названия, более общая. Она позволяет вычислять разные типы кепстральных коэффициентов, и для вычисления мел-частотных кепстральных коэффициентов потребуется предварительно вычислить спектрограмму и применить к ней набор мел-фильтров. Вторая функция, *mfcc*, более проста в использовании, но обладает меньшей гибкостью в вопросе выбора банков фильтров. На вход этой функции можно подавать как отсчеты сигнала во временной области, так и предварительно вычисленную спектрограмму.

Таким образом, мы рассмотрели предпосылки возникновения мел-частотных кепстральных коэффициентов и особенности их вычисления. Традиционными спектрограммами, мел-спектрограммами и МЧКК-граммами не ограничивается весь арсенал признаков, используемых для представления звуковых сигналов. Существует большое количество типов банков фильтров, как универсальных, так и созданных для решения достаточно узких задач. Для примера здесь можно упомянуть гамматонные фильтры, преобразование, которое на английском языке называется Constant-Q transform, банки октавных и третьоктавных фильтров. Пик интереса к этой тематике пришелся на период,

предшествовавший возникновению и массовому распространению концепции глубокого обучения. В настоящее же время значительная часть исследователей сужает выбор до описанных выше типов спектрограмм и достигает высокой эффективности алгоритмов в решаемых задачах, пользуясь преимуществами глубокого обучения.

Кроме рассмотрения непосредственно признаков речевых сигналов, стоит также уделить немного внимания более широкому вопросу, связанному с подготовкой данных для обучения и тестирования нейронной сети. Предположим, есть достаточно большой набор речевых сигналов, который чаще всего называют базой сигналов (dataset). Однако не весь объем сигналов используется для обучения. В современной практике машинного обучения принято разделять весь доступный набор данных на три набора: тренировочный, проверочный (валидационный) и тестовый (контрольный). Это связано с тем, что построение современных алгоритмов практически всегда подразумевает настройку гиперпараметров сети (например, количества слоев и коэффициентов внутри каждого слоя). Многократная проверка работы модели на проверочных данных может негативно сказаться на обобщающих способностях обучаемого нейросетевого алгоритма, то есть на способности алгоритма эффективно работать с новыми, неизвестными для него данными. Именно поэтому подбор гиперпараметров осуществляется на проверочных данных, а финальная оценка возможностей алгоритма – на тестовой выборке. Однократное разбиение данных не является единственно возможным вариантом. Очень часто в условиях ограниченности доступных данных прибегают к их разбиению на блоки с последующей перекрестной проверкой. В любом случае разбиение данных на подвыборки – очень ответственный этап, влияющий как на качество обучения сети, так и на адекватность характеристик, получаемых на стадии тестирования.

Если же изначальный объем доступных данных мал, прибегают к аугментации, то есть к искусственному расширению обучающей выборки. Методы аугментации сильно зависят от типа сигнала и решаемой задачи. Наиболее распространенные практики включают в себя различные типы искажений: зашумление, растягивание и сжатие по временной оси, а также более сложные преобразования сигналов.

Задания для самостоятельного выполнения

1. С использованием языка программирования Python или среды моделирования Matlab осуществите вычисление спектрограмм, мел-спектрограмм, спектрограмм на основе мел-частотных кепстральных коэффициентов для произвольного фрагмента речевого сигнала. Длительность фрагмента лучше выбрать в интервале от одной до нескольких десятков секунд. Проанализируйте, как параметры, задаваемые при вычислении спектрограмм, влияют на результат. За счет чего можно увеличить разрешение по частоте и по времени? Можно ли визуально отличить на спектрограммах интервалы времени, соответствующие согласным и гласным звукам? Как параметры спектрограммы влияют на ее размер?
2. Синтезируйте сигналы, состоящие из фрагментов гармонических колебаний разной частоты и суммы таких колебаний, частотно-модулированных колебаний, последовательностей прямоугольных импульсов и шумоподобных фрагментов. Позволяют ли используемые виды спектрограмм визуально идентифицировать эти типы сигналов? Какой вариант представления можно назвать более удобным и информативным?

2. Оценка и улучшение качества речевых сигналов

Очень часто в реальных условиях речевой сигнал подвергается зашумлению и иным искажениям, ухудшающим качество и разборчивость речи. Для улучшения качества таких сигналов существуют специальные методы и алгоритмы. Однако возможное количество разных типов искажений велико и их характер столь отличен друг от друга, что, как правило, приходится идти по пути создания отдельных методов. Поэтому, кроме методов шумоподавления, существуют также методы деверберации, эхоподавления, восстановления клиппированных сигналов и т. д.

Сравнение существующих методов улучшения качества, а также их разработка невозможна без объективных показателей качества. В отличие от субъективных методов, в которых непосредственная оценка качества осуществляется людьми, объективные можно полностью реализовать программно или аппаратно, что существенно ускоряет и удешевляет процедуру.

Объективные показатели качества принято разделять на эталонные и неэталонные. Для работы последних, в отличиие от эталонных, не требуется чистого (незашумленного, неискаженного) сигнала. Разумеется, создание таких методов является гораздо более сложной задачей. Поэтому долгое время практически все показатели были эталонными. По сути, эти методы измеряют расстояние между чистым и зашумленным сигналами в некотором выбранном пространстве, однако выбор признаков и метрики – сложная научная задача. Самые простые подходы, основанные на измерении отношения сигнал/шум или вычислении среднеквадратической ошибки между сигналами, мало подходят для оценки качества речи. Результаты таких измерений будут слабо коррелировать с субъективными оценками качества, проведенными экспертами. За несколько последних десятилетий тематика объективной оценки качества речи существенно эволюционировала. Некоторым промежуточным, но важным шагом стало создание показателя качества PESQ (Perceptual Evaluation of Speech Quality). Важной особенностью этого метода стал учет психоакустических особенностей восприятия речи человеком. К достоинствам этого метода можно отнести учет влияния широкого спектра искажений, а не только аддитивных шумов.

Развитие нейронных сетей, а также упрощение интерфейсов средств разработки привели к появлению в последние годы значительного количества нейросетевых показателей качества, в том числе

и неэталонных. Для решения этой задачи используют различные архитектуры сетей. В качестве признаков речевых сигналов на входе нейросети чаще всего используют спектрограммы, мел-спектрограммы, а также мел-частотные кепстральные коэффициенты. Последние два типа признаков позволяют учесть ряд психоакустических особенностей восприятия речи. Существенной проблемой является то, что для обучения таких алгоритмов требуются большие базы речевых сигналов с указанными оценками качества, полученными субъективным методом. Процесс записи и разметки речевых баз традиционно считается ресурсоемким, однако для ряда приложений разметка может осуществляться операторами с невысокой квалификацией, а порой сбор и разметку баз можно даже частично автоматизировать. В случае же с оценкой качества процесс подготовки и разметки базы практически не автоматизируется и требует значительных методических и организационных затрат. Однако и эта проблема может быть частично решена. Можно обучить нейронную сеть предсказывать значения PESQ (или другого существующего показателя качества) в неэталонном режиме. В этом случае речевую базу, искаженную различными способами, автоматически размечают с использованием эталонного метода, обладающего высокой достоверностью. Далее эта база используется для обучения неэталонного нейросетевого алгоритма. Примером реализации подобного подхода является алгоритм Quality-Net [1], построенный на основе двунаправленной сети долгой краткосрочной памяти (BLSTM). Подобный подход может также использоваться для усовершенствования процесса обучения нейросетевых алгоритмов шумоподавления [2], что будет рассмотрено далее.

Другим примером использования нейронных сетей в задаче неэталонной оценки качества может служить алгоритм DNSMOS, разработанный сотрудниками компании Microsoft. Для его обучения использовались данные, размеченные в соответствии со стандартом оценки качества ITU-T P.808.

Перейдем к рассмотрению задачи шумоподавления. Большинство алгоритмов шумоподавления осуществляет этот процесс в спектральной области. Такой подход сложился еще до массового применения методов машинного обучения в этой задаче. Многие традиционные алгоритмы шумоподавления строятся на применении функции коррекции спектра (gain function) к спектрограмме зашумленного сигнала, а затем преобразованием результата обработки во временную область

посредством обратного преобразования Фурье. Другая группа алгоритмов строится на применении в спектральной области бинарных масок. Основное отличие алгоритмов состоит в методиках оценки функции коррекции спектра или бинарных масок. Эта оценка может осуществляться и с использованием нейронных сетей, но такой подход является не самым распространённым.

Анализ современных подходов к подавлению шума в речевых сигналах позволяет сделать вывод, что для решения этой задачи возможно применять самые разные виды нейросетей. Простейший алгоритм шумоподавления может быть реализован даже на основе простой полносвязной сети. Наибольшее распространение получили алгоритмы на основе сверточных и рекуррентных нейросетей, однако существуют и менее распространенные подходы, например на основе специально разработанной генеративной сети SEGAN [3] и даже сети WaveNet [4], изначально разработанной для синтеза речи и других звуковых сигналов. Но даже при фиксированном типе нейросети, например сверточной, существует большее количество настраиваемых гиперпараметров, процедур подготовки данных, методик обучения и так далее, что способно породить большое количество исследований и получаемых в ходе них алгоритмов.

Рассмотрим более подробно решение задачи подавления шума с использованием нейронных сетей. Ограничимся наиболее простыми архитектурами сетей, знакомство с которыми не должно вызвать значительных затруднений. Начнем с полносвязной нейронной сети [5]. Важнейшей структурной единицей этой сети является нейрон. Как правило, он обладает несколькими входами и осуществляет взвешенное суммирование входных сигналов с последующей нелинейной обработкой результата суммирования (рис. 4). Выходной сигнал нейрона можно записать в следующем виде:

$$\mathrm{y} = f\left(\sum_{i=1}^{n} w_i x_i + b\right),$$

где x_i – сигнал i-го входа нейрона, w_i – соответствующий весовой коэффициент, b – величина смещения, $f(\cdot)$ – некоторая нелинейная функция, обычно называемая функцией активации.

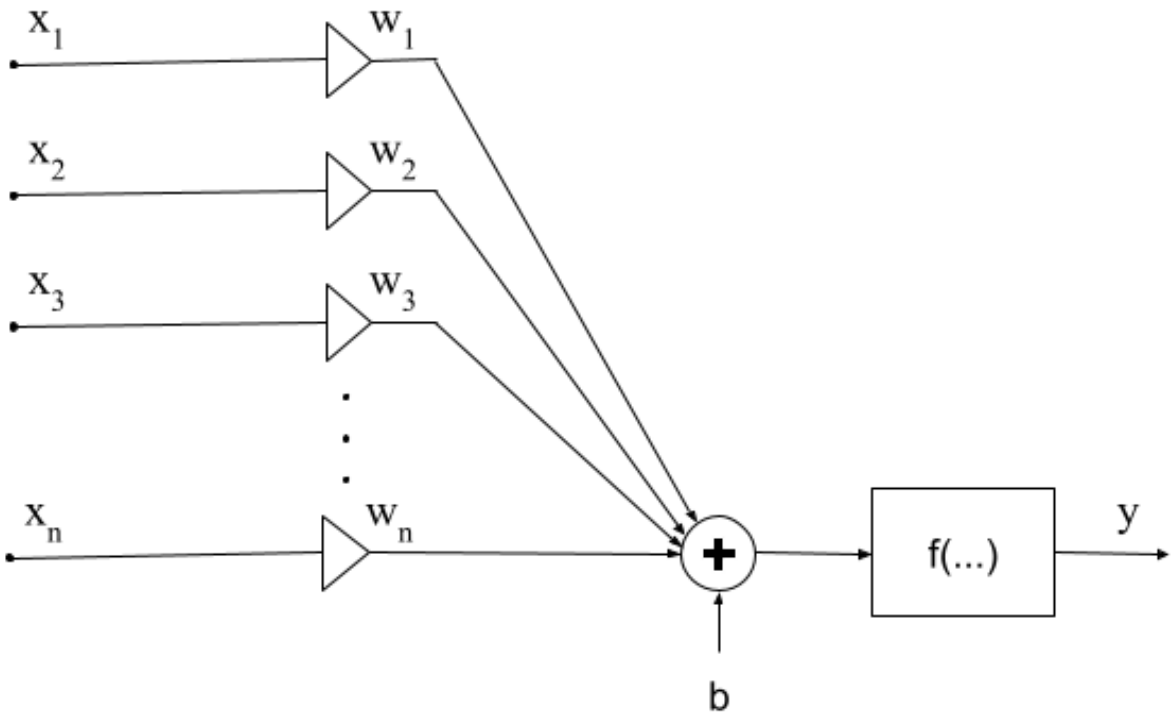

Рис. 4. Структура искусственного нейрона

Примером простой функции активации может служить ступенчатая функция, известная также как функция Хевисайда. В настоящий момент известно большое количество различных функций активации (сигмоида, гиперболический тангенс, обратный квадратный корень и т.д.). В нашем примере воспользуемся функцией ReLU (Rectified Linear Unit) [6], название которой дословно переводится как блок линейного выпрямления (ректификации). Работу этой функции можно описать следующим образом:

$$f(x) = max(0, x).$$

Несмотря на кажущуюся простоту функции активации (рис. 5), ее нелинейный характер во многом определяет возможности нейронных сетей при решении широкого спектра задач обработки и классификации сигналов. Кроме того, существуют усовершенствованные версии этой функции. Например, линейный выпрямитель с утечкой (Leaky ReLU) и параметрический линейный выпрямитель (PReLU).

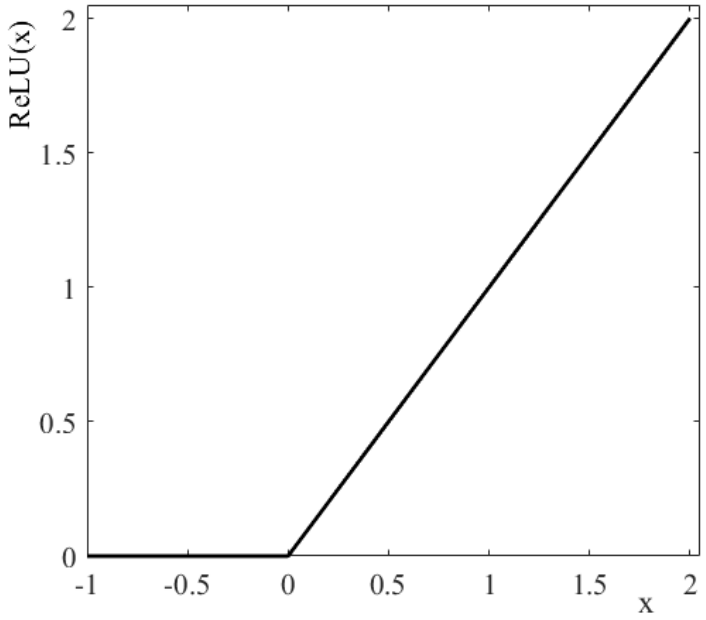

Рис. 5. Функция активации ReLU

Рассмотренный нейрон является наименьшей структурной единицей нейронной сети. Отдельные нейроны объединяются в слои, которые, в свою очередь, и образуют сеть. Выделяют входной, промежуточные (скрытые) и выходной слои (рис. 6). Количество скрытых слоев и количество нейронов в них являются важнейшими гиперпараметрами полносвязной сети.

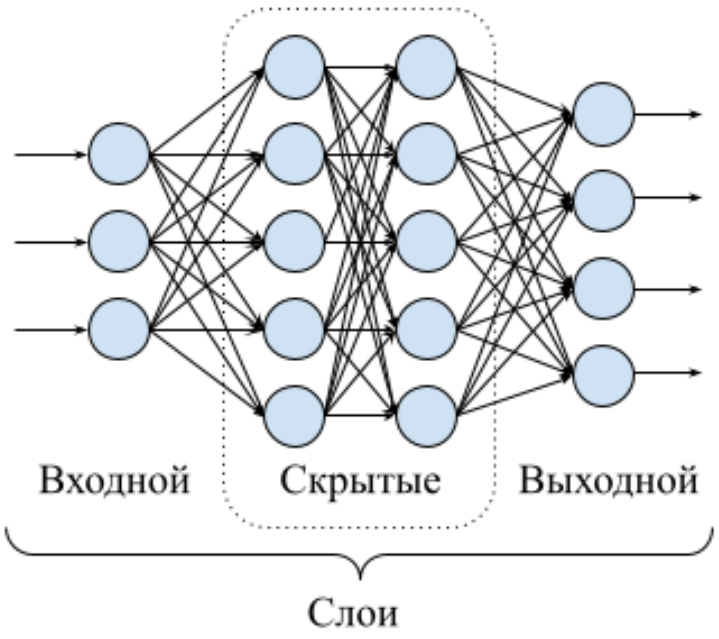

Рис. 6. Структура полносвязной сети с двумя скрытыми слоями

В рассматриваемом примере полносвязная сеть будет иметь два скрытых слоя, каждый из которых будет состоять из 1024 нейронов. В качестве функции потерь выберем среднеквадратическую ошибку, которая будет вычисляться между выходными векторами нейронной сети $\tilde{y}_i$ и незашумленными фрагментами речевых сигналов y_i, используемых при обучении:

$$E = \frac{1}{N}\sum_{i=1}^{N}(y_i - \tilde{y}_i)^2.$$

Функция потерь позволяет оценивать работу нейронной сети на каждом этапе обучения и с помощью алгоритма обратного распространения ошибки производить коррекцию коэффициентов. Выбор функции потерь определяется особенностями решаемой задачи и может оказывать значительное влияние на итоговый результат обучения.

В качестве признаков речевого сигнала будем использовать спектрограммы, полученные с использованием оконного преобразования Фурье. Для частоты дискретизации 8 кГц выберем длину окна 256 отсчетов, коэффициент перекрытия – 0,75, тип окна – окно Хэмминга. При данных параметрах фрагмент сигнала, соответствующий одному окну, описывается 129 спектральными признаками (симметричная часть, отвечающая отрицательным частотам, не используется). Для более точной оценки спектральных свойств шума на вход нейросети лучше подавать не только обрабатываемый вектор, но и несколько предыдущих.

Достаточно распространённой практикой является центрирование и нормализация данных, используемых для обучения нейронной сети. То есть данные приводятся к нулевому математическому ожиданию и единичному стандартному отклонению. В противном случае могут возникнуть проблемы с обучением сети, которое основано на градиентных методах.

После того как определены признаки речевых сигналов, структура и основные параметры сети, выбрана функция потерь, необходимо определиться с методом оптимизации и параметрами процедуры обучения.

Наиболее известным методом оптимизации, применяемом для обучения сетей, является алгоритм стохастического градиентного спуска (Stochastic Gradient Descent, SGD). Существует множество разновидностей и модификаций этого метода, среди которых можно выделить метод адаптивной оценки моментов (Adaptive Moment Estimation, ADAM), пользующийся большой популярностью.

Остановимся на выборе этого метода. Как и для многих других оптимизаторов, основным параметром для него является коэффициент, отвечающий за скорость обучения. В алгоритме ADAM есть и другие параметры, которые обычно не меняют, используя значения, предложенные разработчиками: $\beta_1 = 0{,}9$, $\beta_2 = 0{,}999$, $\epsilon = 10^{-8}$ [7].

Длительность процесса обучения традиционно измеряется эпохами. Каждая эпоха соответствует полному прохождению через сеть всех примеров, составляющих обучающую выборку. В рассматриваемом случае для первого запуска обучения можно ограничиться тремя эпохами. Также перед запуском процесса обучения нужно определиться с разбиением обучающей выборки на фрагменты (пакеты данных), которые принято называть батчами (batch). Размер батча оказывает значительное влияние на скорость и итоговый результат процесса обучения. Для обучения сети выберем размер батча, равный 128. Начальное значение параметра алгоритма ADAM, отвечающего за скорость обучения, будем умножать на 0,9 при переходе к каждой новой эпохе. Это достаточно распространённая практика. Например, в Matlab управлять скоростью обучения можно благодаря функции *traningOptions*, относящейся к Deep Learning Toolbox. В библиотеке Keras такая возможность тоже предусмотрена.

После того как все параметры определены, можно запустить обучение сети. Алгоритм обучения корректирует параметры сети, чтобы минимизировать значение функции потерь на обучающем множестве. По завершении каждой эпохи значение среднеквадратической ошибки также проверяется на сигналах, не входящих в обучающее множество. Если уменьшение значений функции потерь на обучающем множестве сопровождается ростом значений на проверочном множестве, то говорят о переобучении, то есть недостаточной обобщающей способности сети. Существует множество способов борьбы с переобучением, одним из наиболее очевидных является расширение обучающей выборки.

Полносвязные нейронные сети, пример которой мы разобрали выше, стали важным этапом в эволюции машинного обучения, но в настоящее время они уступили место более современным архитектурам и типам сетей, некоторые из которых могут все же содержать полносвязные слои.

Значительный рост популярности нейронных сетей в наше время связан с применением сверточных сетей. Сверточными принято называть сети, состоящие из сверточных слоев, которые получили свое название в силу того, что в их основе лежит математическая операция свертки. Такой слой состоит из набора ядер, также называемых фильтрами.

Исторически большинство достижений в этой области касались задач компьютерного зрения, поэтому в качестве обрабатываемого сигнала обычно подразумевают изображение. Следовательно, ядро является, как минимум, двумерным. В обработке звуковых сигналов используют ядра как с одномерной сверткой, так и с двумерной, поскольку спектрограмма речевого сигнала мало чем отличается от обычного изображения – это такая же матрица, двумерный массив чисел.

Ядро представляет собой окно, состоящее из обучаемых весов, которое, сканируя спектрограмму, производит операцию свертки (рис. 7). В обработке изображений чаще всего используют небольшие размеры окна: 3×3 и 5×5. В обработке звуковых сигналов за счет возможности использования одномерной свертки вариативность значений этого параметра несколько выше. Важно отметить, что чаще всего сверточный слой состоит из нескольких фильтров, что позволяет осуществлять более сложную обработку. Так же как и в случае с полносвязным слоем, сигнал с выхода сверточного слоя поступает на вход функции активации.

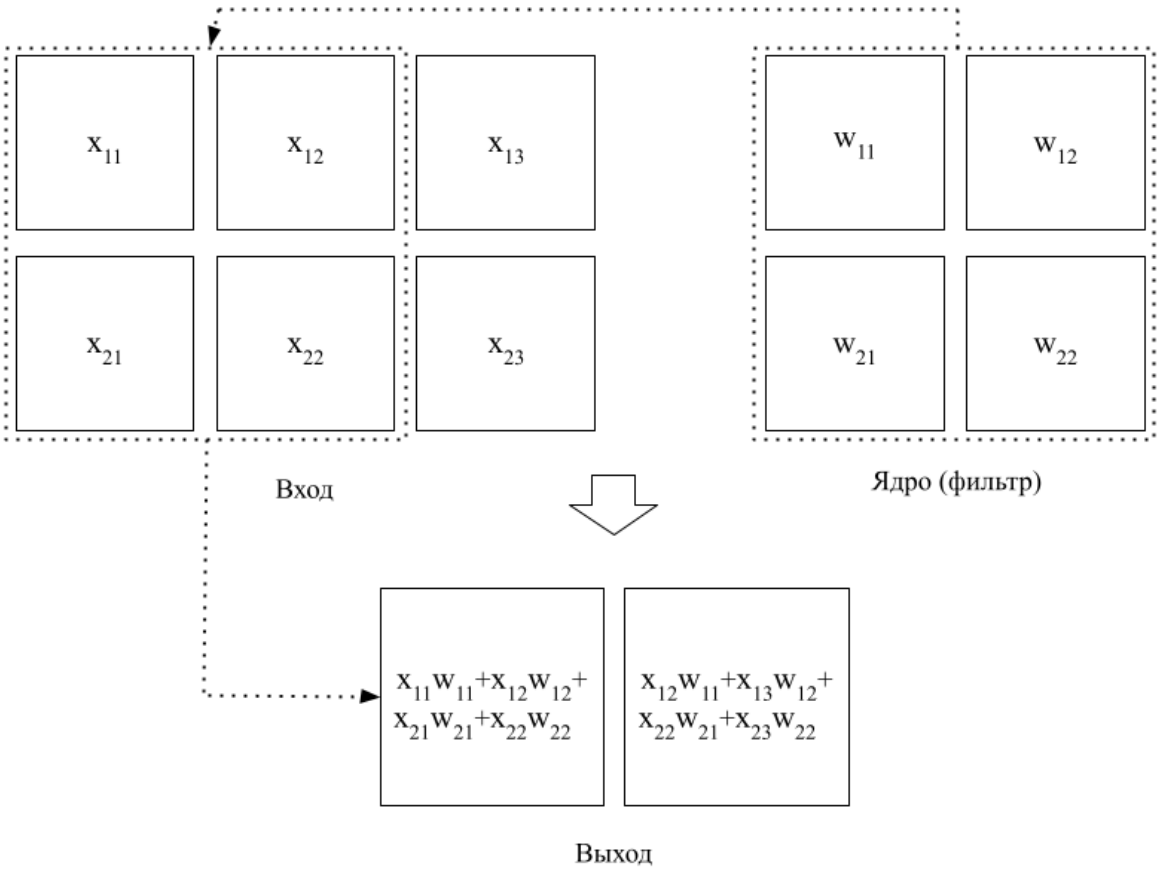

Рис. 7. Пример вычисления двумерной свертки

Достаточно часто, но не всегда вслед за сверточным слоем ставят слой субдескритизации, позволяющий уменьшить размеры обрабатываемого массива. Одним из наиболее популярных является алгоритм субдескритизации, основанный на выборе максимального значения из соседних (max-pooling).

Другой важной для сверточных сетей процедурой, обычно выделяемой в качестве отдельного слоя, является пакетная нормализация (batch normalization). Ее применение положительно влияет на процесс обучения сети [8]. Она состоит из двух основных этапов:

– нормализации по пакету за счет вычитания математического ожидания и деления на среднеквадратическое отклонение,
– умножения результата на константу γ с добавлением β (эти параметры являются обучаемыми, то есть меняются в процессе обучения вместе другими коэффициентами сети).

Рассмотрев основные принципы работы сверточных слоев, можно реализовать на их основе нейросеть, осуществляющую подавление шума в речевых сигналах [9]. Выберем признаки речевых сигналов и процедуру их обработки идентичными случаю полносвязной сети. Аналогично можно поступить и с процедурой обучения. Количество слоев в случае сверточной сети будет больше, чем в предыдущем рассмотренном случае, однако общее число обучаемых параметров при этом сократится. Опишем архитектуру сети, задав основные параметры сверточных слоев (табл. 1). Сеть состоит из 16 сверточных слоев. Первые 15 слоев являются результатом пятикратного повторения 3 слоев с количеством фильтров 18, 30, 8 и размером фильтров 9, 5, 9 соответственно. Для этих слоев используется пакетная нормализация и функция активации ReLU. Заключительный слой имеет один фильтр размером 129.

Стоит заметить, что полученные с помощью этих сетей результаты не могут рассматриваться как демонстрация современного уровня развития науки в области подавления шума в речевых сигналах с использованием нейросетей. Данные примеры призваны дать общее представление о принципах работы нейронных сетей и построении алгоритмов шумоподавления на их основе. Для достижения результатов, отвечающих текущему уровню развития технологий, требуется более глубокое погружение в мир нейронных сетей и машинного обучения.

Однако, не прибегая к детальному рассмотрению современных нейросетевых алгоритмов шумоподавления и принципов их обучения, все же можно познакомиться с перспективными направлениями развития в этой области.

Таблица 1

Параметры сверточных слоев сети

Номер слоя	Количество фильтров	Размер фильтров
1	18	9×8
2	30	5×1
3	8	9×1
4	18	9×1
5	30	5×1
6	8	9×1
7	18	9×1
8	30	5×1
9	8	9×1
10	18	9×1
11	30	5×1
12	8	9×1
13	18	9×1
14	30	5×1
15	8	9×1
16	1	129×1

Традиционно важным шагом при построении алгоритмов шумоподавления является выбор признаков речевых сигналов, подаваемых на вход сети. Как правило, используются спектрограммы, вычисленные с помощью быстрого преобразования Фурье, однако значительная часть исследований посвящена применению и более сложных входных признаков, учитывающих передовые достижения психоакустики. С появлением концепции глубокого обучения, как уже отмечалось, вопросам построения входных признаков стали уделять меньше внимания. Нейронная сеть с большим количеством слоев в процессе обучения сама выделяет необходимые для решения конкретной задачи признаки. Поэтому создатели значительной части современных алгоритмов шумоподавления предпочитают традиционные спектрограммы более сложным в вычислении признакам. Возможно ли дальнейшее упрощение признаков? Ответом на этот вопрос может служить построение блока вычисления признаков, обучаемого вместе с сетью. Такой подход позволяет минимизировать число вычислительных операций, а в некоторых случаях и повысить качество шумоподавления. Примером использования такого подхода может служить алгоритм WaveCRN [10], в котором входной обучаемый сверточный слой заменяет

вычисление спектрограммы зашумленного сигнала. Этот же алгоритм демонстрирует и пример использования слоев, существенно отличных от рассмотренных выше – полносвязных и сверточных. В основе сети лежат рекуррентные слои типа SRU (Simple Recurrent Unit) [11].

Повысить эффективность работы нейросетевого алгоритма шумоподавления можно не только за счет аугментации данных, выбора входных признаков, архитектуры нейросети, но и за счет использования более сложных функций потерь. Ранее уже упоминалась возможность построения нейросетевого неэталонного показателя качества на примере сети Quality-Net [2]. В случаях когда в роли целевого показателя эффективности работы системы выступает один из объективных показателей качества, созданный через обучение, показатель качества может выступать в роли функции потерь. Разумеется, не любая архитектура нейросети подходит для использования в роли функции потерь (требования к ней определяются используемыми методами градиентного поиска и обратного распространения ошибки). Однако такой подход реализован, исследован и позволяет получить ощутимый выигрыш. К его преимуществам можно отнести относительную универсальность – его можно применить к разным алгоритмам шумоподавления, адаптировать для разных целевых показателей качества.

В последние годы все больше внимания в самых разных направлениях глубокого обучения уделяется вопросам трансферного обучения (transfer learning). Такой подход применим и в задаче подавления шума в речевых сигналах. Входные слои нейросети можно обучить на очень большой базе речевых сигналов, например, для решения задачи классификации на большое количество классов. Спецификация задачи, решаемой в рамках первичного обучения, определяется прежде всего доступной базой и особенностями ее разметки. Далее обученные слои могут быть замороженные, а последующие слои обучаются на меньшей базе для решения задачи шумоподавления. Такой подход позволяет обойти проблему доступности специализированных речевых баз, ограниченности вычислительных ресурсов при обучении и добиться высокого качества шумоподавления [12]. Стоит также отметить, что для дообучения можно использовать сеть, обученную другими исследователями, обладающими большими вычислительными ресурсами и доступом к большим речевым базам. Кроме того, такой подход позволяет в короткие сроки находить решения специфических задач.

Кроме вышеозначенных актуальных направлений, можно обозначить проблему адаптации нейросетевых алгоритмов для их использования в компактных устройствах, обладающих ограниченными вычислительными ресурсами, а также попытки создания более универсальных алгоритмов, совмещающих шумоподавление с уменьшением других типов искажений.

Задания для самостоятельного выполнения

1. С использованием языка программирования Python или среды моделирования Matlab реализуйте два алгоритма шумоподавления: на основе полносвязных и сверточных слоев. Можно пользоваться всеми доступными библиотеками функций и любыми другими имеющимися наработками.
2. Произведите обучение нейросетей, используя значения параметров по умолчанию. Что можно сказать о количестве обучаемых параметров каждой из сетей и их скорости обучения?
3. Исследуйте влияние изменения гиперпараметров сетей и внесения в них других допустимых изменений на общее количество обучаемых параметров, скорость и результат обучения. Например, в полносвязной сети можно менять количество нейронов в скрытых слоях, а также исследовать вариант с одним скрытым слоем.
4. Используя реализацию показателя качества PESQ, исследуйте обученные алгоритмы шумоподавления при работе с разными типами шумов, задавая разные значения отношения сигнал/шум. Произведите сравнение двух алгоритмов.

3. Распознавание и синтез речевых сигналов

Рассмотрим задачи, связанные с распознаванием и синтезом речевых сигналов, в которых применение современных типов нейронных сетей дало положительный эффект.

Распознавание речевых сигналов. Под распознаванием речевых сигналов чаще всего подразумевают преобразование речи в текст. Именно такой вариант распознавания имеет наибольшую коммерческую востребованность, а вместе с тем является наиболее сложным. Исходя из особенностей структуры речи, долгое время основные надежды в решении этой задачи были связаны с непосредственным применением аппарата скрытых марковских моделей, однако на современном этапе доминирующую роль играют именно нейросетевые алгоритмы. Для решения задачи распознавания обучаются две вероятностные модели: акустическая и языковая. Первая необходима для того, чтобы для каждого небольшого фрагмента сигнала на основе вычисленных из него признаков определить, какой фонеме он соответствует. Вторая на основе последовательности фонем (с соответствующими им вероятностями) формирует последовательности слов. На этом этапе, благодаря накопленной при обучении статистике, исправляется значительная часть ошибок, возникших при распознавании отдельных фонем.

В конце 1980-х коллектив авторов, включающий в себя Александра Вайбеля и Джеффри Хинтона, предложил архитектуру нейронной сети с временной задержкой (Time Delay Neural Network, TDNN) и продемонстрировал ее применение в задаче распознавания фонем [13]. Эта структура похожа на полносвязную сеть, но нейрон осуществляет взвешенное суммирование не только отсчетов входного сигнала, но и их задержанных копий (рис. 8). Учитывая математическое описание работы такой сети, ее можно отнести к одномерным сверточным сетям. Не лишним будет также заметить, что линейная часть этой сети очень похожа на банк нерекурсивных цифровых фильтров.

Изобретение нейронной сети с временной задержкой по мнению многих специалистов речевой обработки стало отправной точкой в применении нейронных сетей для распознавания речевых сигналов. Эта простая структура до сих пор применяется в речевой обработке, например в задаче распознавания дикторов.

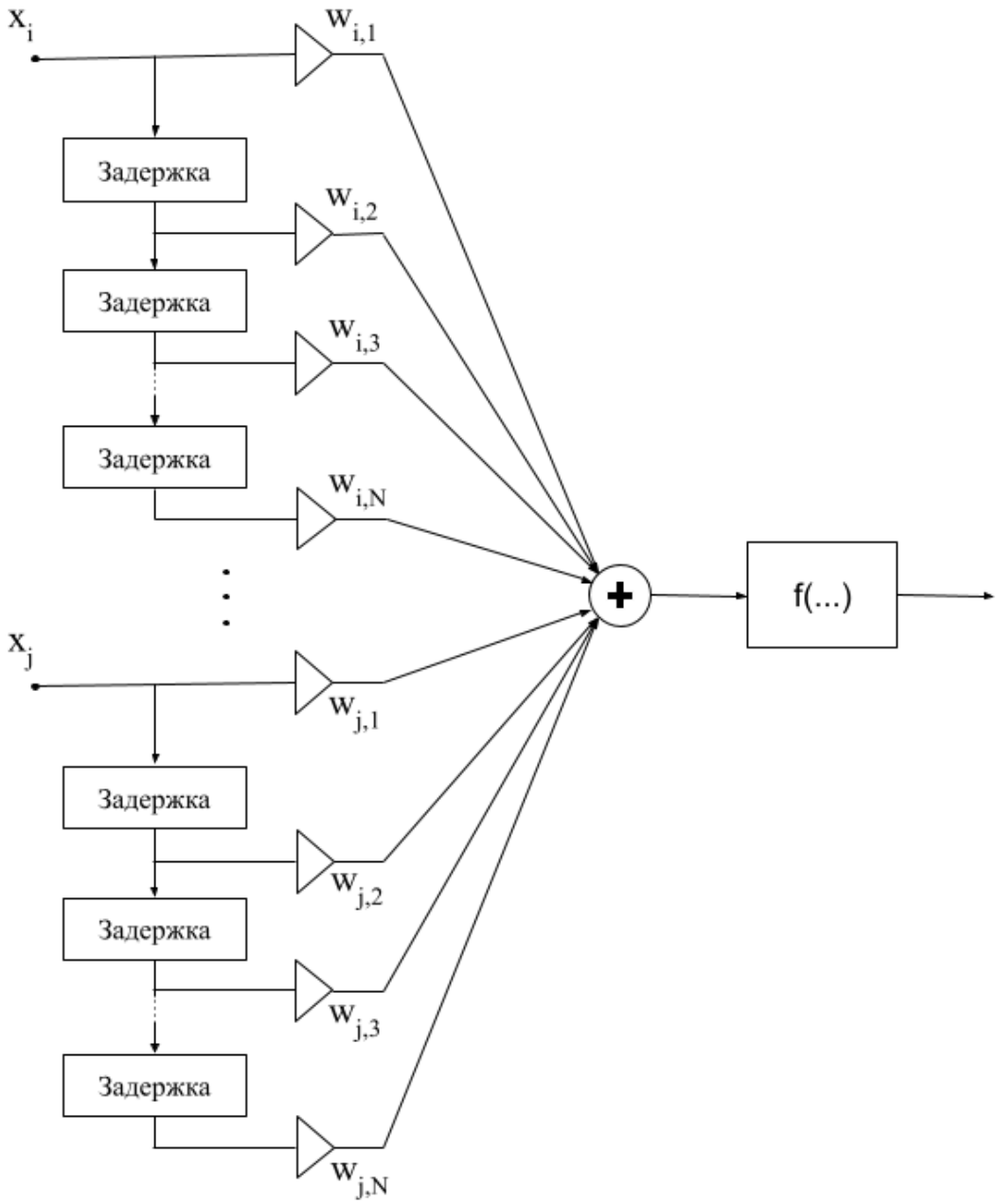

Рис. 8. Схема структурного блока нейронной сети
с временной задержкой

Чтобы оценить современный уровень технологий в задаче распознавания речи, можно обратиться к решениям ИТ-гигантов: Google Cloud Speech-to-Text и Yandex SpeechKit. Оба варианта демонстрируют высокий уровень точности распознавания в благоприятных условиях, однако при работе с живыми записями, сделанными в условии воздействия значительных шумов, помех и реверберации, точность распознавания значительно снижается. Именно неудовлетворительная работа в таких сложных условиях является как основным фактором, сужающим потенциальные области применения алгоритмов

распознавания речи, так и основным научным вызовом в данной области исследований.

Кроме преобразования речи в текст, также можно выделить задачу распознавания отдельных слов (команд) или, например, поиска ключевого слова. Эти задачи можно отнести к более простым, хотя проблема снижения точности в сложной помеховой обстановке также является наиболее значимой. Кроме того, так как эти задачи зачастую являются вспомогательными, служебными для решения какой-то более сложной задачи, особое внимание уделяется снижению потребляемых во время их решения ресурсов.

Распознавание дикторов. К этой группе относятся прежде всего задачи идентификации и верификации говорящего (диктора), а также задача диаризации. Задача идентификации состоит в определении личности по голосу из множества зарегистрированных в системе дикторов. Регистрация подразумевает запись некоторого, обычно короткого, фрагмента голоса конкретного человека. В современных системах хранится не запись голоса, а отклик используемой системы, например нейронной сети, на это считаемое эталонным произнесение. Если все потенциальные дикторы зарегистрированы в системе, то говорят о закрытой задаче идентификации. Если допускается, что сигнал на входе системы может принадлежать незарегистрированному в системе диктору, то говорят об открытой задаче. В этом случае система должна дополнительно детектировать неизвестных дикторов.

Задача верификации направлена на подтверждение личности с помощью сравнения некоторой эталонной записи в базе с текущим произнесением диктора. То есть система верификации отвечает на вопрос, является ли диктор тем, за кого себя выдает.

Как задача идентификации, так и задача верификации могут производиться текстозависимо или текстонезависимо. В первом случае диктору требуется произнести определенное слово или фразу, во втором – что-то произвольное. Текстозависимые системы идентификации более устойчивы к спуфингу – попыткам злоумышленника воспользоваться записью чужого голоса для обмана системы. Другим распространённым подходом противодействия спуфингу является использование многофакторной (мультимодальной) биометрии, когда идентификация производится не только по голосу, но и по другим признакам личности, например по изображению лица или отпечаткам пальцев.

Задача диаризации состоит в разделении звукового потока на фрагменты, принадлежащие разным дикторам. Работу алгоритмов

диаризации часто иллюстрируют осциллограммой звуковой записи, участки которой раскрашены разными цветами, при этом каждый цвет соответствует определенному диктору.

Долгое время основным инструментом решения задач данной группы являлось применение моделей гауссовой смеси (Gaussian Mixture Models или GMM) и других методов, являющихся развитием этого подхода. Попытки применить нейронные сети начали предприниматься достаточно давно, однако существенный прогресс связан с появлением крупных баз сигналов, необходимых для обучения и тестирования глубоких нейронных сетей. Наиболее известным в этом направлении является проект VoxCeleb [14], в рамках которого созданы базы VoxCeleb 1 и VoxCeleb 2. Если же говорить об архитектурах сетей и о специфических подходах, то в первую очередь стоит выделить системы распознавания дикторов на основе x-векторов [15]. Этот подход базируется на применении нейронных сетей с временной задержкой, которые изначально были созданы для распознавания фонем и основаны на одномерной свертке. Альтернативой является применение сверточных сетей, использующих двумерную свертку. Такой тип сетей чаще используется для распознавания изображений, однако двумерность спектрограмм, в том числе и тех, что составлены из МЧКК-векторов, позволяет без существенных изменений использовать известные архитектуры из области машинного зрения для работы с речевыми сигналами. Другим интересным подходом в этой области является замена входного слоя сверточной сети на специально разработанный слой SincNet [16], представляющий собой банк обучаемых фильтров, осуществляющих обработку сигналов, представленных во временной области. Это позволяет не только сократить число вычислений при работе обученной сети, но и повысить точность, а также ускорить процесс обучения.

Кроме вышеупомянутых задач, существует ряд относительно родственных, связанных с анализом голоса конкретных дикторов. Например, существуют алгоритмы определения пола и возраста говорящего, а также эмоциональной окрашенности речи. Все больший интерес вызывают и медицинские приложения речевой обработки, в которых нейронные сети позволяют на основе записей голоса выявить (с определенной вероятностью) ряд патологий.

Синтез речевых сигналов. Попытки синтезировать отдельные звуки, слоги и даже слова осуществлялись еще во времена доминирования аналоговых технологий. С появлением цифровых

технологий ряд подходов были адаптированы и продолжали применяться. Существенная же роль нейросетей в решении этой группы задач начала наблюдаться менее десяти лет назад. В 2016 году компанией Google была представлена нейросетевая архитектура WaveNet [17], которая позволила существенно улучшить качество синтезируемой речи по сравнению с подходами, используемыми ранее. Кроме того, возможности WaveNet не ограничивались исключительно синтезом речевых сигналов. Разработчики также продемонстрировали возможность использования сети для синтеза музыкальных фрагментов. Позже исследователи из разных стран предложили альтернативные подходы и архитектуры нейронных сетей, особенности которых позволяют говорить об отсутствии монопольного положения WaveNet в задачах, связанных с синтезом речи. В качестве примера можно назвать генеративные нейросети WaveGlow, MelGAN, HiFi-GAN [18-20].

Кратко рассмотрим базовые подходы к синтезу речи и возможности использования нейронных сетей в них. Принято выделять конкатенативный (concatenative) и параметрический речевой синтез. Первый строится на сопоставлении коротких фрагментов аудиозаписей, соответствующих отдельным звукам или слогам. Такой подход требует большого количества записанных фрагментов и является крайне негибким. Параметрический подход, как правило, подразумевает синтез речевых сигналов путем последовательного использования акустической модели, которая отвечает также за длительности и интонации, и вокодера, генерирующего звуковые сигналы. Акустическая модель может строиться на основе рекуррентной нейронной сети, однако простые решения на их основе, неспособные учитывать мультимасштабный контекст, демонстрируют низкую реалистичность синтезируемой речи. Пытаясь повысить качество синтезируемой речи, некоторые разработчики комбинируют сильные стороны двух подходов, а для поиска подходящих звуковых фрагментов, заимствованных из конкатенативного подхода, используют нейронные сети. Однако и этот подход не лишен недостатков и не позволяет добиться реалистичных и приятных для восприятия фрагментов речи средней и большой длительности. Поэтому в последние годы наблюдается существенный интерес к применению нейронных сетей в рамках параметрического подхода. При этом нейронные сети используются как для построения акустической модели, так и в качестве вокодера. Например, архитектура Tacotron 2, использует вокодер на основе сети WaveNet. Подобные решения в сочетании с рядом

вспомогательных алгоритмов позволяют уже сейчас синтезировать речь, обладающую высоким уровнем реалистичности.

Чтобы оценить современный уровень технологий в решении этой задачи, можно обратиться к решениям ИТ-гигантов: Google Cloud Text-to-Speech и Yandex SpeechKit. Оба решения демонстрируют высокий уровень естественности синтезируемой речи, который был недоступен еще пять-десять лет назад.

Рассмотренные задачи распознавания и синтеза речи тесно связаны с так называемой обработкой естественного языка. Примером успешного сочетания речевых технологий и обработки естественного языка являются голосовые помощники (например, Алиса, Siri, Alexa). Развитие подобных технологий демонстрирует значительный прогресс в области искусственного интеллекта, достигнутый в последнем десятилетии.

Задания для самостоятельного выполнения

1. С использованием языка программирования Python или в среды моделирования Matlab, а также API Google Cloud Speech-to-Text и Yandex SpeechKit реализуйте системы распознавания коротких речевых сигналов. Произведите сравнение двух решений.
2. Зашумите набор речевых сигналов, состоящий из записей отдельных слов. Проведите исследование точности распознавания сигналов для разных значений отношения сигнал/шум.
3. Используя любой из доступных алгоритмов шумоподавления, произведите обработку зашумленных сигналов из предыдущего задания и проанализируйте, как изменилась точность распознавания. Сделайте выводы.
4. Воспользуйтесь любой из доступных систем синтеза речи по тексту. Насколько высока реалистичность синтезируемой речи? Можно ли ее отличить от естественной? Благодаря чему это можно сделать?

Контрольные вопросы

1. Что такое мел-шкала?
2. Как вычисляются мел-спектрограммы?
3. В чем преимущество мел-спектрограмм перед обычными спектрограммами?
4. Как вычисляются мел-частотные кепстральные коэффициенты?
5. Почему при создании нейросетевых алгоритмов принято разбивать данные на тренировочную, проверочную и тестовую (контрольную) подвыборки?
6. Что такое аугментация данных?
7. Что называется полносвязной нейронной сетью?
8. Что называется эпохой в контексте обучения нейронных сетей?
9. Почему при обучении нейронных сетей градиентными методами прибегают к разбиению данных на пакеты (батчи)?
10. Что называется переобучением?
11. Что из себя представляет сверточный слой?
12. Что называется ядром (фильтром) в контексте сверточных нейронных сетей?
13. Для чего применяют слои субдискретизации?
14. Из каких этапов состоит пакетная нормализация?
15. Какие перспективные подходы используются при разработке современных алгоритмов шумоподавления?
16. Какие роли выполняют акустическая и языковая модели в составе систем распознавания речи?
17. Что из себя представляет нейронная сеть с временной задержкой?
18. В чем отличие задач идентификации и верификации диктора?
19. В чем состоит задача диаризации?
20. В чем основное различие конкатенативного и параметрического речевого синтеза?

Список использованных источников

1. Fu S. W., Tsao Y., Hwang H. T., Wang H. M. Quality-Net: An end-to-end non-intrusive speech quality assessment model based on BLSTM // Conference of the International speech communication association (INTERSPEECH). Hyderabad, India. 2018. P. 1873–1877.
2. Fu S. W., Liao C. F., Tsao Y. Learning with learned loss function: speech enhancement with quality-net to improve perceptual evaluation of speech quality // IEEE signal processing letters. 2019. V. 27. P. 26–30.
3. Pascual S., Bonafonte A., Serra J. SEGAN: Speech enhancement generative adversarial network // Conference of the International speech communication association (INTERSPEECH). Stockholm, Sweden. 2017. P. 3642–3646.
4. Rethage D., Pons J., Serra X. A WaveNet for speech denoising // 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Calgary, Canada. 2018. P. 5069–5073.
5. Liu D., Smaragdis P., Kim M. Experiments on deep learning for speech denoising // Conference of the International speech communication association (INTERSPEECH). Singapore. 2014. P. 2685–2689.
6. Nair V., Hinton G. E. Rectified linear units improve restricted Boltzmann machines // International conference on machine learning (ICML). Haifa, Israel. 2010. P. 807–814.
7. Kingma D. P., Ba J. ADAM: a method for stochastic optimization // 3rd International conference for learning representations. San Diego, USA. 2015. 15 p.
8. Ioffe S., Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift // International conference on machine learning. Lille, France. 2015. P. 448–456.
9. Park S. R., Lee J. A fully convolutional neural network for speech enhancement // Conference of the International speech communication association (INTERSPEECH). San Francisco, USA. 2016. P. 1993–1997.
10. Hsieh T. A., Wang H. M., Lu X., Tsao Y. WaveCRN: An efficient convolutional recurrent neural network for end-to-end speech enhancement // IEEE Signal Processing Letters. New-York, USA. 2020. V. 27. P. 2149–2153.

11. Simple recurrent units for highly parallelizable recurrence / Lei T. et al. // Conference on Empirical Methods in Natural Language Processing. Brussels, Belgium. 2018. P. 4470–4481.
12. Germain F. G., Chen Q., Koltun V. Speech denoising with deep feature losses // Conference of the International speech communication association (INTERSPEECH). Graz, Austria. 2019. P. 2723–2727.
13. Phoneme recognition using time-delay neural networks / Waibel A. et al. // IEEE transactions on acoustics, speech, and signal processing. New-York, USA. 1989. V. 37. Is. 3. P. 328–339.
14. Nagrani A., Chung J. S., Zisserman A. Voxceleb: a large-scale speaker identification dataset // Conference of the International speech communication association (INTERSPEECH). Stockholm, Sweden. 2017. P. 2616–2620.
15. X-vectors: robust DNN embeddings for speaker recognition / Khudanpur S. et al. // IEEE international conference on acoustics, speech and signal processing (ICASSP). Calgary, Canada. 2018. P. 5329–5333.
16. Ravanelli M., Bengio Y. Speaker recognition from raw waveform with SincNet // IEEE Spoken language technology workshop (SLT). Athens, Greece. 2018. P. 1021–1028.
17. WaveNet: a generative model for raw audio / Oord A. V. D. et al. // Proc. 9th ISCA Workshop on Speech Synthesis Workshop. Sunnyvale, USA. 2016. P. 125.
18. Prenger R., Valle R., Catanzaro B. WaveGlow: a flow-based generative network for speech synthesis // 2019 IEEE International conference on acoustics, speech and signal processing (ICASSP). Brighton, UK. 2019. P. 3617–3621.
19. MelGAN: Generative adversarial networks for conditional waveform synthesis / Kumar K. [et al] // Advances in neural information processing systems. Vancouver, Canada. 2019. V. 32. P. 14910–14921.
20. Kong J., Kim J., Bae J. HiFi-GAN. Generative adversarial networks for efficient and high fidelity speech synthesis // Advances in neural information processing systems. Vancouver, Canada. 2020. V. 33. P. 17022–17033.

Оглавление

Учебное издание

Топников Артем Игоревич

ПРИМЕНЕНИЕ НЕЙРОННЫХ СЕТЕЙ В ЗАДАЧАХ ОБРАБОТКИ РЕЧЕВЫХ СИГНАЛОВ

Учебно-методическое пособие

Редактор, корректор Л. Н. Селиванова
Верстка А. И. Топников

Подписано в печать 22.11.2022. Формат 60×84$^{1}/_{16}$.
Усл. печ. л. 2,1. Уч.-изд. л. 1,9.
Тираж 3 экз. Заказ .

Оригинал-макет подготовлен
в редакционно-издательском отделе ЯрГУ.

Ярославский государственный университет
им. П. Г. Демидова.
150003, Ярославль, ул. Советская, 14.