

Содержание

Об авторе	12
Предисловие	13
Благодарности	16
Предисловие от издательства	18
Глава 1. Введение	19
1.1. Проблемы, стоящие перед обработкой естественного языка.....	19
1.2. Нейронные сети и глубокое обучение	20
1.3. Глубокое обучение в NLP	21
1.3.1. Истории успеха.....	22
1.4. Состав и организация материала.....	24
1.5. Что не вошло в книгу	27
1.6. Замечание о терминологии.....	27
1.7. Математическая нотация	28
Часть I. КЛАССИФИКАЦИЯ С УЧИТЕЛЕМ И НЕЙРОННЫЕ СЕТИ ПРЯМОГО РАСПРОСТРАНЕНИЯ	29
Глава 2. Основы обучения и линейные модели	30
2.1. Обучение с учителем и параметрические функции.....	30
2.2. Обучающий, тестовый и контрольный наборы	31
2.3. Линейные модели	33
2.3.1. Бинарная классификация	33
2.3.2. Логлинейная бинарная классификация	37
2.3.3. Многоклассовая классификация.....	38
2.4. Представления	39
2.5. Представления в виде унитарного и плотного векторов	40
2.6. Логлинейная многоклассовая классификация	41
2.7. Обучение как оптимизация.....	41
2.7.1. Функции потерь.....	43
2.7.2. Регуляризация	46
2.8. Градиентная оптимизация.....	47
2.8.1. Стохастический градиентный спуск.....	48
2.8.2. Полный пример.....	50
2.8.3. Не только СГС	52

Глава 3. От линейных моделей к многослойным перцептронам	53
3.1. Ограничения линейных моделей: проблема XOR	53
3.2. Нелинейные преобразования входа	54
3.3. Ядерные методы	54
3.4. Обучаемые отображающие функции	55
Глава 4. Нейронные сети прямого распространения	56
4.1. Метафора, инспирированная мозгом.....	56
4.2. Математическая нотация	58
4.3. Репрезентативная способность.....	60
4.4. Стандартные нелинейности	60
4.5. Функции потерь	62
4.6. Регуляризация и прореживание	62
4.7. Слои вычисления сходства и расстояния.....	63
4.8. Слои погружения.....	64
Глава 5. Обучение нейронной сети	65
5.1. Абстракция графа вычислений	65
5.1.1. Прямое вычисление	67
5.1.2. Вычисление на обратном проходе (производные, обратное распространение)	67
5.1.3. Программное обеспечение.....	68
5.1.4. Рецепт реализации.....	71
5.1.5. Композиция сети.....	72
5.2. Практические вопросы	72
5.2.1. Выбор алгоритма оптимизации.....	72
5.2.2. Инициализация	72
5.2.3. Перезапуск и ансамбли.....	73
5.2.4. Исчезающие и взрывающиеся градиенты	74
5.2.5. Насыщение и мертвые нейроны.....	74
5.2.6. Тасование.....	75
5.2.7. Скорость обучения	75
5.2.8. Мини-пакеты.....	75
Часть II. РАБОТА С ДАННЫМИ ЕСТЕСТВЕННОГО ЯЗЫКА	76
Глава 6. Признаки для текстовых данных	77
6.1. Типология проблем классификации NLP	77
6.2. Признаки для проблем NLP	79
6.2.1. Непосредственно наблюдаемые свойства.....	80

6.2.2. Производные лингвистические свойства.....	83
6.2.3. Базовые и комбинационные признаки	87
6.2.4. N-граммные признаки	87
6.2.5. Дистрибутивные признаки	88

Глава 7. Примеры признаков в NLP

7.1. Классификация документов: определение языка.....	90
7.2. Классификация документов: тематическая классификация	91
7.3. Классификация документов: установление авторства.....	91
7.4. Слово в контексте: частеречная разметка.....	92
7.5. Слово в контексте: распознавание именованных сущностей	94
7.6. Слово в контексте, лингвистические признаки: разрешение лексической многозначности предлогов.....	95
7.7. Отношения между словами в контексте: анализ методом разложения на дуги	97

Глава 8. От текстовых признаков к входным данным

8.1. Кодирование категориальных признаков.....	100
8.1.1. Унитарное кодирование	100
8.1.2. Плотные погружения (погружения признаков).....	101
8.1.3. Плотные векторы и унитарные представления.....	101
8.2. Объединение плотных векторов.....	103
8.2.1. Оконные признаки	103
8.2.2. Переменное число признаков: непрерывный мешок слов.....	104
8.3. Соотношение между унитарным и плотным векторами.....	105
8.4. Разные разности.....	106
8.4.1. Дистанционные и позиционные признаки	106
8.4.2. Дополнение, неизвестные слова и прореживание слов	107
8.4.3. Комбинации признаков.....	108
8.4.4. Обобществление векторов	109
8.4.5. Размерность.....	110
8.4.6. Словарь погружений	110
8.4.7. Выход сети	110
8.5. Пример: частеречная разметка.....	111
8.6. Пример: анализ методом разложения на дуги	112

Глава 9. Языковое моделирование.....

9.1. Задача языкового моделирования.....	115
9.2. Оценивание языковых моделей: перплексивность	116
9.3. Традиционные подходы к языковому моделированию.....	117
9.3.1. Для дальнейшего чтения	118
9.3.2. Ограничения традиционных языковых моделей	118
9.4. Нейросетевые языковые модели	119
9.5. Использование языковых моделей для порождения	123
9.6. Побочный продукт: представления слов.....	124

Глава 10. Предобученные представления слов	125
10.1. Случайная инициализация.....	125
10.2. Специализированное предобучение с учителем.....	125
10.3. Предобучение без учителя	126
10.3.1. Использование предобученных погружений.....	127
10.4. Алгоритмы погружения слов.....	128
10.4.1. Дистрибутивная гипотеза и представления слов	128
10.4.2. От нейросетевых языковых моделей к распределенным представлениям.....	133
10.4.3. Объединяя миры	136
10.4.4. Другие алгоритмы.....	137
10.5. Выбор контекстов.....	138
10.5.1. Подход на основе окон.....	138
10.5.2. Предложения, абзацы или документы	139
10.5.3. Синтаксическое окно	139
10.5.4. Многоязычные контексты.....	141
10.5.5. Представления на основе литер и подслов	141
10.6. Обработка многословных единиц и словоизменения.....	142
10.7. Ограничения дистрибутивных методов	143
Глава 11. Использование погружений слов	146
11.1. Получение векторов слов	146
11.2. Сходство слов.....	147
11.3. Кластеризация слов.....	147
11.4. Нахождение похожих слов.....	147
11.4.1. Сходство с группой слов	148
11.5. Вычеркивание лишних	148
11.6. Сходство коротких документов	148
11.7. Словесные аналоги	149
11.8. Донастройка и проекции.....	150
11.9. Практические вопросы и подводные камни.....	151
Глава 12. Пример: применение архитектуры прямого распространения для вывода смысла предложения	152
12.1. Естественно-языковой вывод и набор данных NLI	152
12.2. Сеть для установления сходства текстов	154
Часть III. СПЕЦИАЛИЗИРОВАННЫЕ АРХИТЕКТУРЫ	157
Глава 13. Детекторы n-грамм: сверточные нейронные сети	159
13.1. Свертка + пулинг – основы	161
13.1.1. Одномерная свертка текста.....	161

13.1.2. Пулинг векторов	163
13.1.3. Вариации	166
13.2. Альтернатива: хеширование признаков	166
13.3. Иерархические свертки	167

Глава 14. Рекуррентные нейронные сети:

последовательности и стеки	171
14.1. Абстракция РНС	172
14.2. Обучение РНС	174
14.3. Типичные примеры использования РНС	175
14.3.1. Приемщик	175
14.3.2. Кодировщик	176
14.3.3. Преобразователь	176
14.4. Двухнаправленные РНС (biRNN)	177
14.5. Многослойные РНС	179
14.6. РНС для представления стеков	180
14.7. Замечание о чтении литературы	182

Глава 15. Конкретные архитектуры рекуррентных нейронных сетей

184	184
15.1. SVOW как РНС	184
15.2. Простая РНС	184
15.3. Вентильные архитектуры	185
15.3.1. LSTM	187
15.3.2. GRU	188
15.4. Другие варианты	189
15.5. Прореживание в РНС	190

Глава 16. Моделирование с помощью рекуррентных сетей

192	192
16.1. Приемщики	192
16.1.1. Классификация по эмоциональной окраске	192
16.1.2. Определение грамматической правильности согласования глагола с субъектом	194
16.2. РНС как экстракторы признаков	196
16.2.1. Частеречная разметка	196
16.2.2. Классификация документов с применением РНС-СНС	199
16.2.3. Анализ зависимостей методом разложения на дуги	199

Глава 17. Условная генерация

202	202
17.1. РНС-генераторы	202
17.1.1. Обучение генераторов	203

17.2. Условная генерация (кодировщик–декодер).....	203
17.2.1. Модели типа последовательность–в–последовательность	205
17.2.2. Приложения	207
17.2.3. Другие обуславливающие контексты.....	209
17.3. Установление сходства предложений без учителя.....	210
17.4. Условная генерация с вниманием	212
17.4.1. Вычислительная сложность	215
17.4.2. Возможность интерпретации	215
17.5. Модели на основе внимания в NLP	215
17.5.1. Машинный перевод.....	215
17.5.2. Морфологическое словоизменение	217
17.5.3. Синтаксический анализ	218

Часть IV. ДОПОЛНИТЕЛЬНЫЕ ТЕМЫ

219

Глава 18. Моделирование деревьев с помощью рекурсивных нейронных сетей

220

18.1. Формальное определение.....	221
18.2. Обобщения и вариации	223
18.3. Обучение рекурсивных нейронных сетей.....	224
18.4. Простая альтернатива – линеаризованные деревья.....	224
18.5. Перспективы.....	225

Глава 19. Предсказание структурного выхода.....

226

19.1. Структурное предсказание на основе поиска.....	226
19.1.1. Структурное предсказание с помощью линейных моделей.....	226
19.1.2. Нелинейное структурное предсказание	227
19.1.3. Вероятностная целевая функция (CRF)	229
19.1.4. Приближенный поиск.....	230
19.1.5. Переранжирование	230
19.1.6. Смотрите также	231
19.2. Жадное структурное предсказание	231
19.3. Условная генерация как предсказание структурного выхода.....	232
19.4. Примеры	233
19.4.1. Структурное предсказание на основе поиска: анализ зависимостей первого порядка	233
19.4.2. Нейросетевые CRF для распознавания именованных сущностей.....	235
19.4.3. Аппроксимация CRF в задаче NER лучевым поиском.....	238

Глава 20. Обучение каскадное, многозадачное и с частичным привлечением учителя

240

20.1. Каскадирование моделей	241
20.2. Многозадачное обучение	244

20.2.1. Обучение в многозадачной конфигурации.....	246
20.2.2. Избирательное обобществление.....	246
20.2.3. Предобучение погружений слов как многозадачное обучение.....	247
20.2.4. Многозадачное обучение в условной генерации.....	248
20.2.5. Многозадачное обучение как регуляризация.....	248
20.2.6. Подводные камни.....	248
20.3. Обучение с частичным привлечением учителя.....	249
20.4. Примеры.....	250
20.4.1. Предсказание взгляда и сжатие предложений.....	250
20.4.2. Пометка дуг и синтаксический разбор.....	251
20.4.3. Разрешение лексической многозначности предлогов и предсказание перевода предлогов.....	252
20.4.4. Условная генерация: многоязычный машинный перевод, синтаксический анализ и описание изображений.....	253
20.5. Перспективы.....	254
Глава 21. Заключение.....	255
21.1. Что мы узнали?.....	255
21.2. Что ждет впереди?.....	255
Список литературы.....	257

Об авторе

Йоав Гольдберг занимается обработкой естественного языка более десяти лет. Он работает старшим преподавателем на факультете информатики университета имени Бар-Илана в Израиле. До этого был исследователем в компании Google Research, Нью-Йорк. Получил степень доктора философии по информатике и обработке естественного языка в университете имени Бен-Гуриона в 2011 году. Регулярно участвует в мероприятиях по NLP и машинному обучению, является членом редколлегии журнала *Computational Linguistics*. Автор более 50 научных работ, получал премии за лучшую и выдающуюся работу на крупных конференциях по обработке естественного языка. В сферу научных интересов входят машинное обучение естественному языку, структурное предсказание, синтаксический анализ, обработка языков с развитым морфологическим строем, а в последние два года еще и нейросетевые модели с упором на рекуррентные нейронные сети.

Предисловие

Обработка естественного языка (Natural Language Processing – NLP) – термин, относящийся к различным способам вычислительной обработки человеческих языков. Сюда входят как алгоритмы, принимающие на входе созданный человеком текст, так и алгоритмы, порождающие тексты, которые выглядят как естественные. Потребность в таких алгоритмах постоянно растет: люди каждый код производят все возрастающие объемы текстов и ожидают, что компьютер будет общаться с ними на их родном языке. При этом задачи, возникающие при обработке естественного языка, очень сложны, поскольку человеческим языкам присуща неоднозначность, изменчивость, да и определены они нестрого.

Естественный язык по природе своей символичный, и первые попытки обработать его тоже были основаны на работе с символами: логике, правилах и онтологиях. Но крайняя неоднозначность и непостоянство языка потребовали алгоритмических подходов, в большей степени основанных на статистике. В настоящее время наиболее распространенные методы основаны на *статистическом машинном обучении*. Уже больше десяти лет в NLP преобладают подходы, основанные на линейных моделях обучения с учителем, такие как перцептроны, метод опорных векторов и логистическая регрессия. Модели обучаются на векторах очень высокой размерности, но при этом крайне разреженных.

Примерно в 2014 году в отрасли наметился переход от линейных моделей над разреженными входными данными к нелинейным нейросетевым моделям над плотными данными. Некоторые нейросетевые методы являются обобщениями линейных моделей и могут использоваться вместо них в линейных классификаторах. Другие, более продвинутые, требуют изменить взгляд на задачу, а взамен предлагают новые возможности моделирования. В частности, подходы, основанные на рекуррентных нейронных сетях (РНС), опираются на марковское предположение, характерное для последовательностных моделей, и позволяют иметь в качестве условий сколь угодно длинные последовательности, порождая при этом эффективные экстракторы признаков. Эти успехи привели к прорывам в языковом моделировании, автоматическом машинном переводе и других приложениях.

При всей своей эффективности нейросетевые методы по различным причинам задают высокий порог входа. В этой книге я сделал попытку познакомить как уже работающих в области NLP специалистов, так и новичков с предпосылками, терминологией, инструментами и методиками, необходимыми для понимания принципов, лежащих в основе нейросетевых моделей языка, чтобы они могли применять их в своей работе. Я также надеюсь познакомить специалистов по машинному обучению и нейронным сетям с предпосылками, терминологией, инструментами и образом мыслей, необходимыми для эффективного применения их знаний к работе с языковыми данными.

Наконец, льщу себя надеждой, что эта книга послужит сравнительно доступным (пусть и неполным) введением в NLP и машинное обучение для тех, кто только начинает изучение той или другой дисциплины.

Предполагаемая аудитория

Эта книга рассчитана на читателей, имеющих техническую подготовку в области информатики или смежных с ней областях знаний и желающих поскорее освоить нейросетевые методы обработки естественного языка. Хотя предполагается, что основными читателями книги будут студенты магистратуры, специализирующиеся на обработке лингвистической информации и машинном обучении, я старался сделать книгу полезной и для уже сложившихся исследователей в области NLP или машинного обучения (путем включения дополнительных материалов), а также для тех, у кого еще нет опыта работы с машинным обучением или NLP (благодаря построению изложения с основ данной темы). Но последней группе читателей, очевидно, придется работать усерднее.

Хотя книга самодостаточна, я все же предполагаю у читателей знание математики, в частности теории вероятностей, алгебры и математического анализа в объеме младших курсов университета, а также базовое владение алгоритмами и структурами данных. Знакомство с машинным обучением было бы очень полезно, но не обязательно.

Эта книга родилась из обзорной статьи [Goldberg, 2016], существенно расширенной и несколько реорганизованной с целью сделать ее полнее и осветить некоторые темы, по разным причинам не вошедшие в обзор. В книге также имеется гораздо больше конкретных примеров приложения нейронных сетей к лингвистическим данным, чем в обзоре. Кроме того, книга рассчитана на читателей, не имеющих подготовки в области NLP или машинного обучения, тогда как в обзоре предполагается наличие соответствующих знаний. Вообще, читатели, знакомые с обработкой естественного языка в том виде, в каком она практиковалась между 2006 и 2014 годом, т. е. существенно опирающейся на машинное обучение и линейные модели, вероятно, сочтут, что журнальный вариант короче и лучше организован. Но и такие читатели могут извлечь пользу из глав о погружении слов (10 и 11), из главы об условной генерации с помощью РНС (17) и из глав, посвященных структурному предсказанию и многозадачному обучению (19 и 20).

Задачи этой книги

Книга задумана самодостаточной, представляющей различные подходы в рамках единой системы обозначений и методики. Однако основное ее назначение – служить введением в механизм нейронных сетей (глубокого обучения) и их применение к лингвистическим данным, а не углубленное изложение основ теории машинного обучения и технологии обработки естественного языка. Отсылаю читателя к другим ресурсам, если такая необходимость возникнет.

Аналогично книга не является исчерпывающим источником для тех, кто намерен развивать нейросетевые подходы (хотя и может служить неплохой отправной точкой). Скорее, она рассчитана на читателей, интересующихся современным состоянием технологии и поиском полезных и нестандартных путей ее применения к стоящим перед ними задачам обработки языка.



Для дальнейшего чтения. Общее всестороннее обсуждение нейронных сетей, стоящей за ними теории, продвинутых методов оптимизации и других тем следует искать в других источниках. В особенности рекомендую книгу Bengio et al. [2016]¹.

Доступное, хотя и строгое введение в практическое машинное обучение имеется в бесплатной публикации Daumé III [2015], которую я горячо рекомендую. Теоретические вопросы машинного обучения прекрасно освещены в бесплатной книге Shalev-Shwartz and Ben-David [2014]² и в учебнике Mohri et al. [2012].

Строгое введение в NLP см. в книге Jurafsky and Martin [2008]. В книге по информационному поиску Manning et al. [2008] также имеется информация, относящаяся к работе с языковыми данными.

Наконец, желающим получить базовые лингвистические знания рекомендуем книгу Bender [2013], вышедшую в этой же серии. Она содержит лаконичное, но достаточно полное изложение вопроса, ориентированное на любителей вычислительных методов. Стоит также познакомиться с первыми главами вводного учебника Sag et al. [2003] по грамматике языков.

На момент написания этой книги исследования в области нейронных сетей и глубокого обучения идут полным ходом. Сказать, что такое текущее состояние дел, невозможно, и я не могу надеяться на сохранение актуальности. Поэтому моя задача – охватить в основном уже устоявшиеся, надежные методы, доказавшие свою работоспособность в нескольких приложениях, а также несколько избранных методов, которые еще нельзя считать полноценными, но которые я считаю либо вошедшими в практику, либо многообещающими.

Йоав Гольдберг
Март 2017

¹ *Йошуа Бенджио, Ян Гудфеллоу, Аарон Курвилль. Глубокое обучение. М.: ДМК Пресс, 2017.*

² *Шай Шалев-Шварц, Шай Бен-Давид. Идеи машинного обучения. М.: ДМК Пресс, 2019.*

Благодарности

Эта книга выросла из моей обзорной статьи [Goldberg, 2016], которая, в свою очередь, появилась вследствие моего недовольства отсутствием четко организованного и ясно изложенного материала о пересечении глубокого обучения с обработкой естественного языка – темы, которую я тогда пытался изучить сам и преподавать своим студентам и коллегам. Поэтому я в долгу перед многими людьми, поделившимися своими замечаниями об обзоре (в различных формах – от замечаний по начальным редакциям до комментариев после публикации), а также теми, кто следил за ходом работы над рукописью книги. Кто-то делился замечаниями лично, кто-то – по электронной почте, а кто-то – в случайных беседах в Твиттере. На книгу также оказали влияние люди, которые не комментировали ее (а иногда даже и не читали), но обсуждали со мной затронутые в ней темы. Одни из них – специалисты по глубокому обучению, другие – по NLP, третьи – по тому и другому, а четвертые пытались изучать эти дисциплины. Кто-то (немногие) присылал очень подробные комментарии, другие обсуждали мелкие детали, а кто-то занимает промежуточную позицию. Но все они повлияли на окончательный текст книги. Перечисляю их в алфавитном порядке: Йоав Арци (Yoav Artzi), Йонатан Ауманн (Yonatan Aumann), Джейсон Бэлбридж (Jason Baldridge), Мигель Баллестерос (Miguel Ballesteros), Мохит Бансал (Mohit Bansal), Марко Барони (Marco Baroni), Тал Баумель (Tal Baumel), Сэм Боумэн (Sam Bowman), Джордан Бойд-Грабер (Jordan Boyd-Graber), Крис Брокетт (Chris Brockett), Минь Вэй Чанг (Ming-Wei Chang), Дэвид Чيانг (David Chiang), Кююн Хуэн Чо (Kyunghyun Cho), Гжегож Хрупала (Grzegorz Chrupala), Александр Кларк (Alexander Clark), Рафаэль Коэн (Raphael Cohen), Райан Коттерелл (Ryan Cotterell), Хэл Дауме III (Hal Daumé III), Николас Дронен (Nicholas Dronen), Крис Дайер (Chris Dyer), Якоб Эйзенштейн (Jacob Eisenstein), Джейсон Эйзнер (Jason Eisner), Майкл Эльхадад (Michael Elhadad), Йад Фазк (Yad Faeq), Манаал Фаруки (Manaal Faruqi), Амир Глоберсон (Amir Globerson), Фредерик Годэн (Frédéric Godin), Эдвард Грэфенштетте (Edward Grefenstette), Мэтью Хоннибал (Matthew Honnibal), Дирк Хови (Dirk Hovy), Моше Коппель (Moshe Koppel), Анджелики Лазариду (Angeliki Lazaridou), Тал Линзен (Tal Linzen), Тханг Луонг (Thang Luong), Крис Мэннинг (Chris Manning), Стивен Мерити (Stephen Merity), Пал Мишель (Paul Michel), Маргарет Митчелл (Margaret Mitchell), Пьеро Молино (Piero Molino), Грэм Нойбиг (Graham Neubig), Иоаким Нивре (Joakim Nivre), Брендан О’Коннор (Brendan O’Connor), Никос Паппас (Nikos Pappas), Фернандо Перейра (Fernando Pereira), Барбара Планк (Barbara Plank), Ана-Мария Попеску (Ana-Maria Popescu), Делип Рао (Delip Rao), Тим Роктэшел (Tim Rocktäschel), Дэн Рот (Dan Roth), Александр Раш (Alexander Rush), Наоми Сафра (Naomi Saphra), Джаме Седдах (Djamé Seddah), Эрел Сегал-Халеви (Erel Segal-Halevi), Ави Шмидман (Avi Shmidman), Шалтиэль Шмидман (Shaltiel Shmidman), Ноа Смит (Noah Smith), Андерс Сёго (Anders Søgaard), Абе Стэнвей (Abe Stanway), Эмма Штрубелль (Emma Strubell), Сандип Субраманиан (Sandeep Subramanian), Ли Лин Тань (Liling Tan), Реут Царфати (Reut Tsarfaty), Питер Тэрни (Peter Turney), Тим Визейра (Tim Vieira), Ориол Виньялс (Oriol Vinyals), Андреас Влачос (Andreas Vlachos), Вэнь Пэн Инь (Wenpeng Yin), Торстен Цэш (Torsten Zesch).

Разумеется, в этот список не вошли очень многие исследователи, с которыми я общался благодаря их публикациям в академических изданиях.

Книга также много приобрела – и даже сформировалась – в результате моего взаимодействия с группой обработки естественного языка в университете имени Бар-Илана (и ее ответвлениях): Йосси Ади (Yossi Adi), Рое Ахарони (Roee Aharoni), Оded Аврахам (Oded Avraham), Идо Даган (Ido Dagan), Джессика Фиклер (Jessica Fidler), Якоб Голдбергер (Jacob Goldberger), Хила Гонен (Hila Gonen), Джозеф Кешет (Joseph Keshet), Элияху Кипервассер (Eliyahu Kiperwasser), Рон Конигсберг (Ron Konigsberg), Омер Леви (Omer Levy), Орен Меламуд (Oren Melamud), Габриэль Становски (Gabriel Stanovsky), Ори Шапира (Ori Shapira), Миха Шлэйн (Micah Shlain), Веред Шварц (Vered Shwartz), Хиллэл Тауд-Табиб (Hillel Taub-Tabib), Рэйчел Уитис (Rachel Witics). Большинство их них входит в оба списка, но я стремился к краткости.

Анонимные рецензенты книги и обзорной статьи – хотя они и не названы (а иногда сильно раздражали) – внесли немало замечаний, предложений и исправлений, которые, безусловно, значительно улучшили окончательную редакцию во многих отношениях. Спасибо всем вам! И отдельная благодарность Грэму Хирсту (Graeme Hirst), Майклу Моргану (Michael Morgan), Саманте Дрейпер (Samantha Draper) и К. Л. Тондо (C.L. Tondo), которые координировали всю работу.

Как обычно, все ошибки – целиком и полностью моя вина. Сообщите мне, если что-то найдете, чтобы внести исправления в следующее издание, если таковое когда-нибудь состоится.

Наконец, хочу поблагодарить свою жену Ноа, которая терпела и поддерживала меня во время писательских загулов, своих родителей Эстер и Авнера и брата Надав, которые зачастую переживали из-за моей идеи написать книгу больше, чем я сам. А также персонал кафе The Streets и кафе Shne'or, которые кормили и поили меня на протяжении всего процесса написания, почти не отвлекая от дела.

Йоав Гольдберг
март 2017

Предисловие от издательства

Отзывы и пожелания

Мы всегда рады отзывам наших читателей. Расскажите нам, что вы думаете об этой книге – что понравилось или, может быть, не понравилось. Отзывы важны для нас, чтобы выпускать книги, которые будут для вас максимально полезны.

Вы можете написать отзыв прямо на нашем сайте www.dmkpress.com, зайдя на страницу книги, и оставить комментарий в разделе «Отзывы и рецензии». Также можно послать письмо главному редактору по адресу dmkpress@gmail.com, при этом напишите название книги в теме письма.

Если есть тема, в которой вы квалифицированы, и вы заинтересованы в написании новой книги, заполните форму на нашем сайте по адресу http://dmkpress.com/authors/publish_book/ или напишите в издательство по адресу dmkpress@gmail.com.

Список опечаток

Хотя мы приняли все возможные меры для того, чтобы удостовериться в качестве наших текстов, ошибки все равно случаются. Если вы найдете ошибку в одной из наших книг – возможно, ошибку в тексте или в коде, – мы будем очень благодарны, если вы сообщите нам о ней. Сделав это, вы избавите других читателей от расстройств и поможете нам улучшить последующие версии этой книги.

Если вы найдете какие-либо ошибки в коде, пожалуйста, сообщите о них главному редактору по адресу dmkpress@gmail.com, и мы исправим это в следующих тиражах.

Нарушение авторских прав

Пиратство в интернете по-прежнему остается насущной проблемой. Издательства «ДМК Пресс» и Ракет очень серьезно относятся к вопросам защиты авторских прав и лицензирования. Если вы столкнетесь в интернете с незаконно выполненной копией любой нашей книги, пожалуйста, сообщите нам адрес копии или веб-сайта, чтобы мы могли применить санкции.

Пожалуйста, свяжитесь с нами по адресу электронной почты dmkpress@gmail.com со ссылкой на подозрительные материалы.

Мы высоко ценим любую помощь по защите наших авторов, помогающую нам предоставлять вам качественные материалы.

Глава 1

Введение

1.1. Проблемы, стоящие перед обработкой естественного языка

Обработка естественного языка – это наука о проектировании методов и алгоритмов, которые принимают или порождают неструктурированные данные естественного языка. Человеческие языки в высшей степени неоднозначны (сравните фразы «Я ел пиццу с друзьями» и «Я ел пиццу с оливками») и вариативны (смысл фразы «Я ел пиццу с друзьями» можно выразить и так: «Мы с друзьями разделили пиццу»). Языки также постоянно изменяются и развиваются. Люди прекрасно справляются с порождением и пониманием языковых конструкций, они способны выразить, воспринять и интерпретировать фразы с утонченным смыслом, изобилующим нюансами. Но, будучи замечательными *пользователями* языка, люди очень плохо справляются с формальным описанием правил, *управляющих* языком.

Поэтому понимание и порождение языка с помощью компьютеров – чрезвычайно трудная задача. Лучшие методы работы с языковыми данными основаны на алгоритмах машинного обучения с учителем, которые пытаются вывести паттерны и закономерности использования из множества предварительно аннотированных пар входных и выходных текстов. Рассмотрим, к примеру, задачу классификации документа по одной из четырех категорий: Спорт, Политика, Светская хроника и Экономика. Очевидно, что содержащиеся в документе слова дают вполне определенные указания, но какое именно указание дает данное слово? Выписать соответствующие правила довольно трудно. Однако читатели легко могут отнести документ к теме, а затем, основываясь на нескольких сотнях аннотированных таким образом документов в каждой категории, алгоритм машинного обучения с учителем может вывести паттерны использования слов, которые помогут классифицировать новые документы. Методы машинного обучения отлично работают в задачах, где определить хороший набор правил очень трудно, а аннотировать входные примеры выходными метками сравнительно просто.

Помимо проблем, связанных с обработкой неоднозначных и вариативных входных данных в системе с плохо определенными и отсутствующими наборами правил, у естественного языка есть и дополнительные свойства, которые еще больше затрудняют разработку вычислительных подходов на основе машинного обучения: *дискретность, композиционность и разреженность*.

Язык по природе своей символический и дискретный. Основными элементами письменного языка являются литеры. Литеры образуют слова, обозначающие предметы, понятия, события, действия и идеи. Литеры и слова – дискретные символы: слова «гамбургер» или «пицца» вызывают определенные мысленные образы, но также являются разными символами, смысл которых отделен от них и интерпретируется в нашем мозгу. Не существует внутренней связи между «гамбургером» и «пиццей», которую можно было бы вывести из самих символов или составляющих их букв. Сравните это с таким понятием, как цвет, играющий важнейшую роль в машинном зрении, или звуковые сигналы: эти величины непрерывны, что позволяет, например, перейти от цветного изображения к тоновому с помощью простой математической операции, или сравнить два цвета на основе внутренне присущих им свойств, например оттенка и яркости. Со словами так не получится – не существует простой операции, которая позволила бы перейти от слова «красный» к слову «розовый» без использования большой справочной таблицы или словаря.

Язык также обладает свойством композиционности: буквы образуют слова, слова образуют фразы и предложения. Смысл фразы может быть больше смысла составляющих ее слов и определяется набором запутанных правил. Поэтому, чтобы интерпретировать текст, приходится подняться над уровнем букв и слов и рассматривать длинные последовательности слов, например предложения или даже полные документы.

Сочетание описанных выше свойств ведет к *разреженности данных*. Число комбинаций слов (дискретных символов), имеющих смысл, практически бесконечно. Число допустимых предложений огромно, нет никакой надежды перечислить их все. Откройте любую книгу – подавляющее большинство встречающихся в ней предложений вы никогда раньше не видели и не слышали. Более того, вполне вероятно, что многие последовательности четырех слов, встречающиеся в этой книге, тоже не попадались вам ранее. Если вы возьмете газету, вышедшую каких-то 10 лет назад, или попытаете вообразить газету, которая выйдет через 10 лет, то многие слова, особенно имена людей, названия торговых марок и компаний, а также жаргонные словечки и технические термины, покажутся незнакомыми. Нет никакого очевидного способа получить одно предложение из другого путем обобщения или определить сходство предложений, не зависящее от их смысла, который для нас является ненаблюдаемой величиной. Поэтому задача обучения на примерах становится крайне трудной: даже располагая гигантским набором примеров, мы все равно с большой вероятностью будем наблюдать события, которые в этом наборе не встречались и сильно отличаются от всех встречающихся примеров.

1.2. Нейронные сети и глубокое обучение

Глубокое обучение – раздел машинного обучения. Это другое название нейронных сетей – семейства методов обучения, исторически появившегося в результате изучения способов вычислений в мозге. Его еще можно охарактеризовать как обучение параметрических дифференцируемых функций¹. Само название «глу-

¹ В этой книге преобладающим является математический, а не основанный на аналогии с мозгом взгляд.

бокое обучение» связано с тем, что многочисленные слои, образованные этими дифференцируемыми функциями, зачастую сцеплены друг с другом.

Хотя машинное обучение в целом можно описать как обучение делать предсказания на основе прошлых наблюдений, в глубоком обучении упор делается на том, чтобы научиться не только делать прогнозы, но и *правильно представлять* данные, чтобы они были пригодны для предсказания. Располагая большим набором соответствий между входами и выходами, мы подаем данные на вход сети, которая последовательно преобразует их, пока на выходе не получится готовое предсказание. Какие преобразования выполнять, сеть обучается на основе соответствий между входами и выходами, стремясь выбирать каждое преобразование так, чтобы было проще соотнести данные с желаемой меткой.

Проектировщик-человек отвечает за разработку архитектуры сети и режима обучения, обеспечение сети надлежащим набором обучающих примеров и выбор подходящего способа кодирования входных данных, а трудную работу по обучению правильному представлению берет на себя сеть при поддержке со стороны сетевой архитектуры.

1.3. Глубокое обучение в NLP

Нейронные сети дают эффективный механизм обучения, чрезвычайно привлекательный для использования в задачах обработки естественного языка. Главный компонент языковой нейронной сети – *слой погружения*, т. е. отображение дискретных символов на непрерывные векторы в пространстве сравнительно небольшой размерности. В результате погружения слова преобразуются из изолированных дискретных символов в математические объекты, над которыми можно производить различные действия. В частности, если за меру расстояния между словами взять расстояние между векторами, то будет проще обобщить влияние одного слова на другое. Такое представление слов векторами сеть находит в процессе обучения. Поднимаясь вверх по иерархии, сеть также обучается комбинировать векторы слов способами, полезными для предсказания. Эта возможность до некоторой степени компенсирует дискретность и разреженность данных.

Существует два основных вида архитектуры нейронных сетей, которые можно по-разному комбинировать: сети прямого распространения и рекуррентные/рекурсивные сети.

Сети прямого распространения, в частности многослойные перцептроны (МСП), позволяют работать с входными данными фиксированного размера или с данными переменного размера, если можно не обращать внимания на порядок элементов. Если загрузить в сеть множество входных компонентов, то она обучится комбинировать их осмысленными способами. МСП можно использовать в тех случаях, где раньше применялась линейная модель. Нелинейность сети, а также возможность интегрировать в нее ранее обученные погружения слов часто приводят к выдающейся точности классификации.

Сверточные нейронные сети – это специализированные архитектуры, отличающиеся способностью выделять локальные паттерны в данных: на вход им подаются данные произвольного размера, а они выделяют осмысленные локальные паттерны, чувствительные к порядку слов, независимо от того, в каком месте входных данных они встречаются. Они очень хорошо справляются с идентифи-

кацией фраз в изъявительном наклонении и идиом заранее ограниченной длины в длинных предложениях или документах.

Рекуррентные нейронные сети (РНС) – это специализированные модели для последовательных данных. Они принимают входную последовательность объектов и порождают вектор фиксированной длины, который подытоживает ее. Смысл слов «подытожить последовательность» зависит от задачи (например, информация, необходимая для ответа на вопрос об эмоциональной окраске предложения, отличается от той, что необходима для ответа на вопрос о его грамматической правильности). Поэтому рекуррентные сети редко используются сами по себе, а их ценность заключается в том, что это допускающие обучение компоненты, которые можно подать на вход другим компонентам сети и обучить совместной работе. Например, выход рекуррентной сети можно подать на вход сети прямого распространения, которая попытается предсказать некоторое значение. Рекуррентные сети являются весьма выразительными моделями для последовательностей и являют собой, пожалуй, самое полезное, что могут предложить нейронные сети обработке языков. Они позволяют отказаться от *марковского предположения*, преобладавшего в NLP в течение нескольких десятилетий, и проектировать модели, в которых условиями могут быть целые предложения. При этом они могут при необходимости учитывать порядок слов и не подвержены проблемам статистического оценивания, проистекающим из разреженности данных. Эта возможность дает заметный выигрыш в *языковом моделировании* – задаче о предсказании вероятности следующего слова в последовательности (или, что то же самое, вероятности предложения), – которое является краеугольным камнем многих приложений NLP. Рекурсивные сети обобщают рекуррентные сети с последовательностей на деревья.

Многие задачи в естественном языке *структурированы*, т. е. нуждаются в порождении сложных выходных структур типа последовательностей или деревьев. Нейросетевые модели пригодны и для этих целей – либо путем адаптации известных алгоритмов структурного предсказания для линейных моделей, либо благодаря использованию новых архитектур, таких как модели последовательность-в-последовательность (кодировщик-декодер), которые мы будем называть в этой книге *моделями условной генерации* (conditioned generation model).

Наконец, многие языковые задачи предсказания связаны друг с другом в том смысле, что знание о том, как решить одну из них, помогает при обучении другим. Кроме того, если *аннотированных учителем* обучающих примеров может не хватать, то уж недостатка в исходных (неаннотированных) текстовых данных точно не наблюдается. Можно ли обучаться на основе неаннотированных данных или результатов обучения родственных задач? Нейронные сети предлагают весьма интересные возможности как для многозадачного обучения (multitask learning – MTL), т. е. обучения на основе результатов для родственных задач, так и для обучения с частичным привлечением учителя (обучения на внешних неаннотированных данных).

1.3.1. Истории успеха

Полносвязные нейронные сети прямого распространения (МСП) в большинстве случаев могут использоваться всюду, где обычно применяется линейный обучаемый. Сюда относятся бинарные и многоклассовые проблемы обучения, а также

более сложные структурные проблемы предсказания. Нелинейность сети, а равно возможность простой интеграции с предобученными погружениями слов, часто ведет к выдающейся верности классификации. В серии работ¹ удалось добиться улучшения результатов синтаксического анализа, всего лишь заменив линейную модель анализатора полносвязной сетью прямого распространения. Прямолинейное применение сети прямого распространения в качестве замены классификатора (обычно в сочетании с использованием предобученных векторов слов) дает преимущества во многих лингвистических задачах, в том числе: основополагающей задаче языкового моделирования², суперразметке с комбинаторными категориальными грамматиками (CCG)³, прослеживании состояния диалога⁴ и предупорядочении для статистического машинного перевода⁵. В работе Yu et al. [2015] продемонстрировано, что многоуровневые сети прямого распространения могут давать конкурентоспособные результаты в задачах классификации по эмоциональной окраске и для ответов на вопросы на основе опубликованных сведений о фактах. В работах Zhou et al. [2015] и Andor et al. [2016] такие сети интегрированы в системы структурного предсказания на основе лучевого поиска, что позволило добиться фантастической верности в задачах синтаксического анализа, разметки последовательностей и других.

Сети со сверточными и пулинговыми слоями полезны для задач классификации, в которых мы ожидаем найти сильные локальные признаки, касающиеся принадлежности к классам, но эти признаки могут встречаться в разных местах входных данных. Например, в задаче классификации документов одна ключевая фраза (или n -грамма) может помочь в определении темы документа [Johnson and Zhang, 2015]. Мы хотели бы обучить систему таким последовательностям слов, которые служат хорошими тематическими индикаторами, не думая о том, в каком месте документа они встречаются. Сверточные и пулинговые слои позволяют моделировать обучение поиску таких локальных позиционно-независимых индикаторов. Сверточно-пулинговая архитектура продемонстрировала многообещающие результаты на многих задачах, включая классификацию документов⁶, категоризацию коротких текстов⁷, классификацию по эмоциональной окраске⁸, классификацию по типу связи между сущностями⁹, обнаружение событий¹⁰, обнаружение перефразирования¹¹, пометку семантических ролей¹², ответы на вопросы¹³, прогнозирование кассового сбора от проката фильмов на основе критических от-

¹ [Chen and Manning, 2014, Durrett and Klein, 2015, Pei et al., 2015, Weiss et al., 2015].

² См. главу 9, а также Bengio et al. [2003], Vaswani et al. [2013].

³ [Lewis and Steedman, 2014].

⁴ [Henderson et al., 2013].

⁵ [de Gispert et al., 2015].

⁶ [Johnson and Zhang, 2015].

⁷ [Wang et al., 2015a].

⁸ [Kalchbrenner et al., 2014, Kim, 2014].

⁹ [dos Santos et al., 2015, Zeng et al., 2014].

¹⁰ [Chen et al., 2015, Nguyen and Grishman, 2015].

¹¹ [Yin and Schütze, 2015].

¹² [Collobert et al., 2011].

¹³ [Dong et al., 2015].

зывов¹, моделирование интересности текста² и моделирование связи между последовательностями литер и метками частей речи³.

В естественном языке часто приходится работать со структурированными данными произвольного размера, например последовательностями и деревьями. Мы хотели бы уметь улавливать закономерности в таких структурах или моделировать сходство между ними. Рекуррентные и рекурсивные архитектуры позволяют работать с последовательностями и деревьями, сохраняя при этом большой объем структурной информации. Рекуррентные сети [Elman, 1990] проектировались для моделирования последовательностей, а рекурсивные [Goller and Küchler, 1996] являются обобщениями рекуррентных сетей для обработки деревьев. Показано, что рекуррентные модели дают прекрасные результаты в языковом моделировании⁴, а также в разметке последовательностей⁵, машинном переводе⁶, грамматическом разборе⁷ и многих других задачах, включая нормализацию зашумленного текста⁸, прослеживание состояния диалога⁹, генерацию ответов¹⁰ и моделирование связей между последовательностями литер и метками частей речи¹¹.

Показано, что рекурсивные модели дают лучшие в отрасли или близкие к лучшим результаты в задачах переранжирования результатов синтаксического анализа на основе грамматики составляющих¹² и зависимостей¹³, анализа повествовательной линии¹⁴, классификации семантических связей¹⁵, распознавания политической идеологии на основе деревьев разбора¹⁶, классификации по эмоциональной окраске¹⁷, классификации по эмоциональной окраске в зависимости от цели¹⁸ и ответов на вопросы¹⁹.

1.4. Состав и организация материала

Книга состоит из четырех частей. Часть I представляет собой введение в основные механизмы обучения, которые будут использоваться в книге: обучение с учителя

¹ [Bitvai and Cohn, 2015].

² [Gao et al., 2014].

³ [dos Santos and Zadrozny, 2014].

⁴ Перечислим несколько заметных работ: Adel et al. [2013], Auli and Gao [2014], Auli et al. [2013], Duh et al. [2013], Jozefowicz et al. [2016], Mikolov [2012], Mikolov et al. [2010, 2011].

⁵ [Irsoy and Cardie, 2014, Ling et al., 2015b, Xu et al., 2015].

⁶ [Cho et al., 2014b, Sundermeyer et al., 2014, Sutskever et al., 2014, Tamura et al., 2014].

⁷ [Dyer et al., 2015, Kiperwasser and Goldberg, 2016b, Watanabe and Sumita, 2015].

⁸ [Chrupala, 2014].

⁹ [Mrkšić et al., 2015].

¹⁰ [Kannan et al., 2016, Sordoni et al., 2015].

¹¹ [Ling et al., 2015b].

¹² [Socher et al., 2013a].

¹³ [Le and Zuidema, 2014, Zhu et al., 2015a].

¹⁴ [Li et al., 2014].

¹⁵ [Hashimoto et al., 2013, Liu et al., 2015].

¹⁶ [Iyyer et al., 2014b].

¹⁷ [Hermann and Blunsom, 2013, Socher et al., 2013b].

¹⁸ [Dong et al., 2014].

¹⁹ [Iyyer et al., 2014a].

лем, МСП, градиентные методы обучения и абстракция графа вычислений для реализации и обучения нейронных сетей. В части II механизмы, описанные в первой части, соединяются с лингвистическими данными и объясняется, как их интегрировать с нейронными сетями. Здесь же обсуждаются алгоритмы погружения слов и дистрибутивная гипотеза, а также подходы к языковому моделированию на основе сетей прямого распространения. В части III речь пойдет о специализированных архитектурах и РНС для моделирования последовательностей и стеков. РНС – основное новшество, разработанное для применения нейронных сетей к лингвистическим данным, часть III посвящена по преимуществу именно им, включая обеспечиваемую ими инфраструктуру условной генерации и модели внимания. Часть IV – собрание различных дополнительных тем: применение рекурсивных сетей для моделирования деревьев, модели структурного предсказания и многозадачное обучение.

Часть I, посвященная основам нейронных сетей, состоит из четырех глав. В главе 2 вводятся основные понятия машинного обучения с учителем, параметрические функции, линейные и логлинейные модели, регуляризация и функции потерь, обучение как оптимизация и градиентные методы обучения. Материал излагается с азов и необходим для чтения последующих глав. Читатели, знакомые с основами теории обучения и градиентным обучением, могут пропустить эту главу. В главе 3 описывается главное ограничение нейронных сетей прямого распространения и МСП, обсуждается определение многослойных сетей, их теоретическая ценность и основные составные части, например нелинейность и функции потерь. В главе 4 вводятся нейронные сети прямого распространения и МСП. Обсуждается определение многослойных сетей, их теоретическая ценность и такие стандартные понятия, как нелинейность и функции потерь. Глава 5 посвящена обучению нейронных сетей. Вводится абстракция графа вычислений, которая позволяет автоматически вычислять градиенты в произвольных сетях (алгоритм обратного распространения), и приводится несколько важных советов и приемов эффективного обучения сети.

Часть II, посвященная лингвистическим данным, состоит из семи глав. В главе 6 описана типология проблем лингвистической обработки и обсуждаются источники информации (признаков), доступные нам при работе с лингвистическими данными. В главе 7 приводятся конкретные примеры, показывающие, как признаки, описанные в предыдущей главе, используются для решения различных задач обработки естественного языка. Читатели, знакомые с лингвистической обработкой, могут пропустить эти две главы. В главе 8 материал глав 6 и 7 соединяется с нейронными сетями и обсуждаются различные способы кодирования лингвистических признаков для подачи на вход нейронной сети. В главе 9 мы познакомимся с задачей языкового моделирования и архитектурой нейросетевой модели языка с прямым распространением. Тем самым будет проложен путь к обсуждению предобученных погружений слов в следующих главах. Глава 10 посвящена распределенным и дистрибутивным подходам к представлению значений слов. Вводится подход к дистрибутивной семантике на основе матрицы слово–контекст, а также алгоритмы погружения слов, вдохновленные нейросетевым языковым моделированием, в частности GloVe и Word2Vec; обсуждается их связь с дистрибутивными методами. В главе 11 речь пойдет об использовании погружения слов вне контекста нейронных сетей. Наконец, в главе 12 представлен пример сети прямого

распространения, ориентированной конкретно на задачу логического вывода на естественном языке.

Часть III, где вводятся специализированные сверточные и рекуррентные архитектуры, насчитывает пять глав. В главе 13 рассматриваются сверточные сети, специализированные для обучения информативным n -граммным паттернам. Обсуждается также альтернативный метод хеш-ядра. Оставшиеся в этой части главы 14–17 посвящены РНС. В главе 14 описывается абстракция РНС для моделирования последовательностей и стеков. В главе 15 рассматриваются конкретные примеры РНС, в том числе простая РНС (известная также как РНС Элмана) и вентильные архитектуры, в том числе долгая краткосрочная память (LSTM) и вентильный рекуррентный блок (GRU). В главе 16 приведены примеры моделирования с помощью абстракции РНС в конкретных приложениях. Наконец, в главе 17 описывается инфраструктура условной генерации – основного метода моделирования в современных системах машинного перевода, – а также моделирование предложений без учителя и многие другие инновационные приложения.

Часть IV представляет собой собрание продвинутых и немагистральных тем и состоит из трех глав. В главе 18 вводится понятие древовидных рекурсивных сетей для моделирования деревьев. Хотя это семейство моделей выглядит весьма привлекательно, оно пока находится в стадии исследований, и убедительные истории успеха – дело будущего. Тем не менее о нем должны знать ученые, собирающиеся развивать технику моделирования и дальше. Читатели, которых в основном интересуют зрелые, доказавшие свою надежность методы, могут пропустить эту главу. Глава 19 посвящена структурному предсказанию. Она носит технический характер. Читатели, специально интересующиеся этой темой или уже знакомые с методами структурного предсказания для линейных моделей или для обработки языка, вероятно, оценят приведенный здесь материал. Остальные могут спокойно пропустить эту главу. Наконец, в главе 20 представлено многозадачное обучение и обучение с частичным привлечением учителя. Нейронные сети предлагают широкие возможности для того и другого. Это важные методы, которые пока находятся в стадии исследований. Но уже существующие приемы реализовать относительно просто, и они приносят реальный выигрыш. С технической точки зрения, глава не особенно сложная, поэтому рекомендуется всем читателям.

ЗАВИСИМОСТИ Большинство глав зависят от предшествующих им. Исключения составляют первые две главы части II, которые не зависят от предыдущих и могут читаться в любом порядке. Некоторые главы и разделы можно пропустить без ущерба для понимания остального материала. К ним относятся раздел 10.4 и глава 11, в которых речь идет о деталях алгоритмов погружения слов и использовании погружений вне контекста нейронных сетей; глава 12, в которой описывается специальная архитектура для работы с набором данных Stanford Natural Language Inference (SNLI), ориентированным на естественно-языковой вывод; глава 13, где описываются сверточные сети. При чтении материала о рекуррентных сетях можно без особого ущерба пропустить главу 15, содержащую подробные сведения о конкретных архитектурах. Главы в части IV по большей части не зависят друг от друга, их можно читать в любом порядке или пропустить вовсе.

1.5. Что не вошло в книгу

Книга посвящена применениям нейронных сетей к задачам обработки языка. Но некоторые разделы лингвистической обработки с помощью нейронных сетей сознательно опущены. Особенно я уделял внимание обработке письменного языка, а речевые данные и звуковые сигналы не получили освещения. В рамках письменного языка я рассматриваю низкоуровневые, относительно корректно поставленные задачи, не вторгаясь в такие области, как диалоговые системы, реферирование документов или вопросно-ответные системы, которые считаю открытыми проблемами. Хотя применение описанных методов, возможно, позволит добиться прогресса в решении этих задач, я не привожу никаких примеров и не вдаюсь в их обсуждение. Точно так же за рамками книги остался семантический анализ. Мультимодальные приложения, связывающие лингвистические данные с другими модальностями, например техническим зрением или базами данных, лишь кратко упомянуты. Наконец, обсуждаются в основном англоязычные тексты, а языки с более развитым морфологическим строем и недостаточным объемом аннотированных ресурсов почти не затрагиваются.

Не рассматриваются также некоторые базовые вопросы. В частности, для качественной лингвистической обработки критически важны *надлежащая оценка* и *аннотирование данных*. Оба вопроса оставлены за рамками книги, но читатель должен знать об их существовании.

Надлежащая оценка включает выбор подходящих метрик для оценивания качества алгоритма на данной задаче, передовые практики, честное сравнение с другими работами, выполнение анализа ошибок и оценку статистической значимости.

Аннотирование данных – это хлеб насущный систем NLP. Без данных мы не сможем обучить модели с учителем. В исследованиях мы очень часто используем «стандартные» аннотированные данные, созданные кем-то еще. Но все равно важно знать об источнике этих данных и принимать во внимание следствия, вытекающие из процесса создания набора. Аннотирование данных – весьма обширная тема, включающая корректную постановку задачи, разработку инструкций по аннотированию, принятие решения о выборе источника аннотированных данных, его охвате и пропорциях представительства классов, хорошее разбиение набора на обучающие и тестовые данные, работу с людьми, занимающимися аннотированием, консолидацию решений, контроль качества аннотаторов и аннотирования и многое другое.

1.6. Замечание о терминологии

Термин «признак» используется для обозначения конкретного элемента лингвистических входных данных: слова, суффикса или метки части речи. Например, в частеречном разметчике первого порядка признаками могут быть «текущее слово, предыдущее слово, следующее слово, предыдущая часть речи». Термином «входной вектор» мы обозначаем конкретное значение входных данных. Это не согласуется с основной массой литературы по нейронным сетям, где слово «признак» перегружено, употребляется в обоих случаях, но преимущественно обозначает элемент входного вектора.

1.7. Математическая нотация

Полужирными заглавными буквами обозначаются матрицы (\mathbf{X} , \mathbf{Y} , \mathbf{Z}), полужирными строчными буквами – векторы (\mathbf{b}). Если имеется последовательность связанных матриц и векторов (например, каждая матрица соответствует одному слою сети), то используются верхние индексы (\mathbf{W}^1 , \mathbf{W}^2). В тех редких случаях, когда необходимо обозначить степень матрицы или вектора, мы употребляем скобки: $(\mathbf{W})^2$; $(\mathbf{W}^3)^2$. Квадратные скобки $[\]$ используются как оператор взятия индекса для векторов и матриц: $\mathbf{b}_{[i]}$ обозначает i -й элемент вектора \mathbf{b} , а $\mathbf{W}_{[i,j]}$ – элемент на пересечении i -го столбца и j -й строки матрицы \mathbf{W} . Если возникает неоднозначность, то мы иногда прибегаем к более привычной математической нотации и пишем b_i для обозначения i -го элемента вектора \mathbf{b} и w_{ij} для обозначения элементов матрицы \mathbf{W} . Символом \cdot обозначается оператор скалярного произведения: $\mathbf{w} \cdot \mathbf{v} = \sum_i w_i v_i = \sum_i \mathbf{w}_{[i]} \mathbf{v}_{[i]}$. Последовательность векторов $\mathbf{x}_1, \dots, \mathbf{x}_n$ обозначается $\mathbf{x}_{1:n}$, а последовательность элементов x_1, \dots, x_n – $x_{1:n}$. Символом $\mathbf{x}_{n:1}$ обозначается инвертированная последовательность: $\mathbf{x}_{1:n}[i] = \mathbf{x}_i$, $\mathbf{x}_{n:1}[i] = \mathbf{x}_{n-i+1}$. Конкатенация векторов обозначается $[\mathbf{v}_1; \mathbf{v}_2]$.

Если явно не оговорено противное, векторы считаются векторами-строками, хотя обычно так не делают. Решение использовать векторы-строки, умножаемые справа на матрицы ($\mathbf{x}\mathbf{W} + \mathbf{b}$), отличается от стандартного – в литературе по нейронным сетям, как правило, используются векторы-столбцы, умножаемые на матрицу слева ($\mathbf{W}\mathbf{x} + \mathbf{b}$). Мы верим, что читатель сможет приспособиться к нотации векторов-столбцов при чтении литературы¹.

¹ Решение использовать векторы-строки продиктовано следующими соображениями: оно согласуется с тем, как входные векторы и диаграммы сетей обычно изображаются в литературе; иерархическая (слоистая) структура сети становится более наглядной, и вход оказывается самой левой, а не вложенной переменной; размерности полносвязной сети указываются в порядке $d_{in} \times d_{out}$, а не $d_{out} \times d_{in}$; эта нотация лучше согласована с реализацией сетей в коде с использованием библиотек для работы с матрицами, например `numpy`.