РАЗРАБОТКА БИБЛИОТЕКИ ДЛЯ ПОЛУЧЕНИЯ ФОНЕТИЧЕСКОЙ ТРАНСКРИПЦИИ ДЛЯ РУССКОГО ЯЗЫКА

Badasyan Alexandra Arsenovna
NIU HSE - NN / Nizhny Novgorod,
Russia
sashabadasyan@icloud.com

Abstract

This work is devoted to the development of a library for obtaining phonetic transcription for the Russian language. The library is based on the literary norm of phonetic transcription for the Russian language and uses symbols of the International Phonetic Alphabet. Transcription considers the allocation of allophones. The resulting library can be used in automatic speech recognition and synthesis tasks.

Keywords: Automatic Speech Recognition; Text To Speech; allophone; International Phonetic Alphabet; Russian language; phoneme; phonetic transcription

1 Введение

На сегодняшний день методы распознавания (Automatic Speech Recognition, далее – ASR) и синтеза речи (Text To Speech, далее – TTS) – одна из бурно развивающихся областей обработки естественного языка. Тем не менее, в области ASR и TTS до сих пор наблюдается большой разрыв между лингвистическими и компьютерными знаниями.

При обучении моделей большую роль играет качество и уровень разметки обучающего корпуса. Практические исследования показали, что модель синтеза работает наилучшим образом, если она обучалась на фонемах, а не графемах [8]. Однако на данный момент в открытом доступе нет фреймворка, который предоставлял бы подробную фонетическую транскрипцию в соответствии с нормой для русского языка. Так, целью исследования стала разработка такой библиотеки.

При создании модуля транскрибирования были использованы правила фонетической транскрипции для русского языка, описанные С. В. Князевым и С.К. Пожарицкой [1]. Транскрипция производилась для текстовых представлений записей устной речи.

2 Фонетическая транскрипция

Транскрибирование устного корпуса может осуществляться разными способами. Чаще всего звуковой материал отражается в виде простой текстовой транскрипции, написанной с учетом орфографических и пунктуационных норм и не отражающей тонкостей произношения. В данной работе была поднята проблема создания фонетической транскрипции, основной целью которой является отражение реального произношения.

Минимальной фонетической единицей является фонема. Фонема – это «абстрактное понятие, включающее в себя представление о возможных вариациях одного звука» [1]. Однако в речи они могут менять свои артикуляторно-акустические характеристики, образуя тем самым аллофоны – модификации фонем [1]. Другими словами, аллофон – это конкретная реализация фонемы в отрывке речи. Фонетическая транскрипция подразумевает графическое описание именно аллофонов, поэтому при создании библиотеки автоматического транскрибирования им будет уделено самое большое внимание.

3 Структура библиотеки транскрибирования

Предложенный фреймворк состоит из 5 шагов: предобработка данных, расстановка ударений, выделение фразовых слов, преобразование «буква-фонема» (ПБФ) и выделение аллофонов.

3.1 Подбор материала и предобработка данных

В качестве материала исследования использовался корпус устной речи RUSLAN [10]. Он содержит более 22 тысяч записей, общей продолжительностью более 31 часа. Данный корпус был выбран, т. к. он был создан специально для задач ASR и TTS, и при этом уже был опробован в реальных исследованиях.

В качестве предобработки данных необходимо было произвести базовые нормализацию и токенизацию данных.

Нормализация включила в себя:

1) обработку цифр (прим.: *89 – восемьдесят девять*) и сокращений (прим.: *т. е. – то есть)* и др. путем их вербализации с выбором верной грамматической формы;
2) обработку аббревиатур (прим.: *МВД – эмвэдэ*), которая заключалась в описании по орфоэпическим нормам без расшифровки;
3) замену *е* на *ё* в некоторых словах;
4) обработку слов с дефисом – дефис сохранялся до этапа расстановки ударений, после чего части до и после дефиса объединялись в один токен (прим.: *книга-то – кни+га-то – кни+гато*);
5) приведение всех букв к нижнему регистру;
6) удаление знаков препинания и двойных пробелов.

Нормализация осуществлялась с помощью специальных библиотек (при обработке цифр и сокращений) и регулярных выражений. Токенизация производилась с помощью регулярных выражений.

3.2 Расстановка ударений

Расстановка ударений в словах – это своего рода фундамент для будущей транскрипции, поскольку место ударного слога определяет сильные и слабые позиции гласных фонем в слове. Для этой задачи был использован модуль-аксентор – модуль автоматического расставления ударений.

Для того, чтобы правильно подобрать аксентор, необходимо учесть особенности анализируемых данных. А именно то, что в устных корпусах, составленных для обучения моделей ASR и TTS, часто намеренно используются сложные, спорные лексемы, в числе которых и омографы (т. е. слове, имеющих несколько вариантов ударения), и имена собственные, и иностранные слова, и нецензурная лексика.

Чаще всего аксенторы строятся на основе словарей и анализе дистрибуции. Такой подход позволяет решить проблему различения омографов. Но несмотря на то, что словарные аксенторы имеют относительно высокую точность, их «знания» ограничены ровно на столько, на сколько ограничен используемый словарь. Другими словами, такой модуль не сможет обрабатывать некоторое количество лексем.

В виду этого выбор фреймворка для автоматической расстановки ударений был основан на следующем условии: кроме словарных методов и анализа дистрибуции он должен использовать нейронные сети, поскольку модули, использующие нейронные сети, могут обрабатывать любые слова. Действительно, точность расстановки ударений с помощью нейронных сетей может быть ниже, чем у словарных аксенторов. Тем не менее, в условиях обработки объемных сложных корпусов, такой подход будет наиболее уместным.

Так, для расстановки ударений был выбран модуль-аксентор StressRNN [5]. В рамках этого модуля расстановка ударений идет в два этапа. Сначала он расставляет ударения в соответствии с грамматическим словарём русского языка А. А. Зализняка и

транскрипциями из подкорпуса устной речи из НКРЯ [9]. Затем, если слово не было найдено в словарях, применяются нейронные сети и делается предсказание.

3.3 Выделение фонетических слов

Согласно литературной норме, не все слова имеют полнозначное ударение. В языке существуют такие зависимые элементы, которые фонетически примыкают к опорному слову, создавая таким образом фонетическое слово. Такие элементами называются клитиками [11]. Клитиками чаще всего выступают короткие слова, принадлежащие к служебным частям речи. Также клитикой может выступать личное местоимение, если оно играет синтаксическую роль подлежащего и находится в контактном положении с предикатом.

Фонетические слова могут образовываться двумя способами: проклитическим или энклитическим. Проклитический способ подразумевает присоединение неполноударного элемента к следующему за ним слову, энклитический – к предшествующему.

Для выделения фонетических слов были использованы синтаксические деревья зависимостей, построенные с помощью библиотеки Spacy [7]. С их помощью выделялись минимальные группы зависимых, которые затем объединялись в один токен, если зависимый член относился к служебной части речи (за некоторыми исключениями). Такой же алгоритм применялся к группе подлежащее – сказуемое, где подлежащее выраженно личным местоимением. Например, в предложении "*Мы бы пошли в парк, если бы распогодилось*" модуль строит следующее дерево зависимостей: *(пошли Мы бы (парк в) (распогодилось (если бы)))*. В результате было выделено 5 фонетических слов: *Мыбы пошли впарк еслибы распогодилось*.

4 Преобразование «буква-фонема» (ПБФ)

Транскрипция, в отличии от орфоэпии, должна отражать реальную норму произношения, сложившуюся в современном языке. К сожалению, произносительную норму невозможно описать только регулярными правилами, поскольку любому живому языку свойственно меняться, адаптироваться и создавать исключения.

Для автоматизации выделения исключений был применен процесс преобразования «буква-фонема» (далее – ПБФ). Термин взят из работы Б. М. Лобанова [2]. ПБФ подразумевает замену букв в слове в соответствии с фонемами, отражающими реальную норму произношения. Такое преобразование обычно неосознанно происходит у носителей языка при чтении.

Согласно ПБФ, в моём фреймворке были выделены три группы правил: общие правила, регулярные исключения и нерегулярные исключения. В отличии от первых двух, последняя группа не поддается формальному описанию. Для нее был составлен специальный список исключений.

В рамках библиотеки процесс ПБФ был осуществлен по следующему алгоритму: (1.1) сначала каждое слово проверялось на наличие его в списке нерегулярных исключений (с последующей заменой на соответствующую цепочку букв, согласно реальной норме произношения); (1.2) если слово не находилось в списке, к нему применялись правила, свойственные регулярным исключениям; (2) затем все слова транслитерировались; (3) последним шагом было применение общих правил ПБФ. Реализация первой группы правил в самом конце обусловлена тем, что не всем графемам соответствует фонема, и наоборот. Наглядным примером является сочетание согласной и мягкого знака, где мягкий знак не имеет аналога в числе фонем, зато означает смягчение согласной. Такие преобразования возможно отобразить только после транслитерации.

4.1 Подбор библиотек для автоматизации процесса ПБФ

Прежде чем приступать непосредственно к ПБФ, необходимо было определить, каким образом будут графически отражаться единицы звука. В первую очередь речь идет о втором этапе ПБФ – транслитерации.

На данный момент в большинстве ресурсов открытого доступа используется фонетическая разметка символами CMUdict [4], которая имеет ряд недостатков. Во-первых, этот словарь был разработан для английского языка, что затрудняет его адаптацию для других языков, в том числе русского. Во-вторых, он использует нестандартные транскрипционные знаки – двухсимвольные, с использованием цифр. Такая запись значительно затрудняет интерпретацию данных (прим.: *диалог – /D0 I A L O0 K/*). В-третьих, он состоит всего из 84 символов и не учитывает генерацию большинства аллофонов и просодию.

Для того, чтобы модель была доступна и понятна как можно большему кругу людей, необходимо, чтобы представление данных было максимально приближено к классической фонетической транскрипции для русского языка. Именно поэтому, было принято решение не использовать CMUdict и обратиться к Международному фонетическому алфавиту (далее – МФА). МФА был создан в роли общепринятого набора символов для однозначного обозначения звуков вне зависимости от цели и языка. МФА легче поддается интерпретации, т. к. он создан на основе латинского алфавита (ср.: *диалог – /dʲɪʁlʷok/*). В то же время МФА дает намного более подробное описание звуков. Так, для транслитерации была выбрана библиотека epitran [6], использующая символы МФА.

4.2 Реализация процесса ПБФ

Первый этап ПБФ, т. е. обработка исключений, проводился на основе регулярных правил и списка нерегулярных исключений. Список исключений включил в себя, например, заимствования, в которых буква «е» не смягчает предшествующий согласный (прим.: *каф**е*** *– каф**э***). В число регулярных правил вошли удаление непроизносимых согласных (прим.: *солн**ц**е – сон**ц**е*), замена окончаний прилагательных *-ого, -его* на *-ово, -ево* (прим.: *мил**ого** – мил**ово***), замена окончаний глаголов *-ться, -тся* на *-ца* (прим.: *готови**ться** – готови**ца***), замена *-что-* на *-што-* и др.

С помощью модуля epitran была произведена транслитерация (второй этап ПБФ). После такого преобразования каждый токен получил вид списка фонем с сохранением порядка следования.

Наконец, последним шагом было применение общих правил ПБФ ко всем транслитерированным токенам. В рамках этого шага были описаны смягчение согласных перед буквой «ь» (прим.: *рол**ь** – /ro**l**/– /ro**lʲ**/*), оглушение согласных, имеющих пару по звонкости-глухости, на конце слова (прим.: *ко**д** – /ko**d**/ – /ko**t**/*) и ассимиляция: полная ассимиляция (прим.: ***сж**ечь – /**sz**et͡ɕ/ – /**zː**et͡ɕ/*) частичная ассимиляция по признаку твердости-мягкости (прим.: *ба**н**тик – /ba**n**tʲik/ – /ba**nʲ**tʲik/*), частичная ассимиляция по признаку звонкости-глухости (прим.: *желе**з**ка – /zelʲe**z**ka/– /zelʲe**s**ka/*).

5 Выделение аллофонов

Общие правила автоматического транскрибирования были сформулированы на основе набора аллофонов, описанного Б. М. Лобановым и другими авторами [3]. Этот набор включает в себя 580 единиц и описывает 18 случаев генерации аллофонов. На основе этого набора аллофонов были сформулированы общие правила автоматического выделения аллофонов. При создании фонетической транскрипции эти правила стали базовыми.

Кроме того, было создано 4 дополнительных функции, описывающие аллофоны, которые не попадают под классификацию Б.М. Лобанова, но при этом существуют в речи на русском языке и отображены в МФА: выделение носового губно-зубного аллофона */m/ – /ɱ/* (*амфора*

– *aɱfərʌ*), выделение аллофона */r/(/rʲ/)* перед глухими согласными и в конце слова – */r̥/(/r̥ʲ/)* (*арфа – aɾfʌ*), выделение аллофона */tsʲ/* перед звонкими согласными – */d̥z̥ʲ/* (*плацдарм – plɐd̥z̥darm*), выделение фрикативного */g/ – /ɣ/*.

Полученная библиотека принимает на вход текст на русском языке и возвращает список аллофонов с соблюдением порядка.

Запись	Расшифровка записи	Транскрипция
wavs/000000.wav	*С тревожным чувством берусь я за перо.*	*s t rʲ ɪ vʷ o ʒ nʸ ɨ m t͡ɕ ʷ ʉ s t v ə m bʲ ɪ rʷ u sʲ j æ z ə pʲ ɪ rʷ o*
wavs/000001.wav	*Кого интересуют признания литературного неудачника?*	*k ə vʷ ɵ ɪ nʲ tʲ ɪ. rʲ ɪ s ɐ̟ jʷ ɐ̟ t pʲ rʲ ɪ z n a nʲ ɪ j æ. lʲ ɪ tʲ ɪ. r ə tʷ u r n ə v ʌ nʲ ɪ. ɐ̟ d a t͡ɕ nʲ ɪ k ʌ*
wavs/000002.wav	*Что поучительного в его исповеди?*	*ʂ tʷ o p ə ɐ̟ t͡ɕ ɨ tʲ e lʲ n ə v ʌ vʲ j e vʷ o i s p ə vʲ e dʲ ɪ*
wavs/000003.wav	*Да и жизнь моя лишена внешнего трагизма.*	*d a i ʒʸ ɨ zʲ nʲ m ɐ j a lʲ ɪ ʂ ə n a + v nʲ e ʂ nʲ e v ʌ t r ɐ gʲ i z m ʌ*
wavs/000004.wav	*Я абсолютно здоров.*	*j æ ə b s ɐ lʲʷ ʉ t n ʌ z d ɐ - rʷ o f*

Таблица 1: Полная обработка текстов на примере расшифровок записей корпуса RUSLAN

6 Заключение

По итогу работы был разработан фреймворк для получения фонетической транскрипции для текстов на русском языке в соответствии с литературной нормой. Стоит подчеркнуть, что созданная библиотека использует символы МФА, что позволило создать более подробную и более интерпретируемую транскрипцию. Тем самым был создан инструментарий, который позволяет использовать при обучении моделей ASR и TTS фонетические единицы. В перспективе на основе разработанной библиотеки планируется создание метрики распределения и сбалансированности фонем в устных корпусах, предназначенных для задач ASR и TTS. Готовый проект доступен на GitHub по ссылке: https://github.com/suralmasha/RuTranscript.

References

[1] Князев С.В., Пожарицкая С.К. Современный русский литературный язык: фонетика, графика, орфография, орфоэпия. – Академический проект, 2005.

[2] Лобанов Б.М. Компьютерный синтез и клонирование речи / Лобанов Б.М., Цирульник Л.И. Минск: Белорусская наука, 2008.–344 с.: ил // SEMANTIC TECHNOLOGY DESIGN NL INTERFACES FOR QUESTION ANSWERING. – 2008.

[3] Лобанов Б.М., Пьорковска Б., Рафалко Я., Цирульник Л.И., Шпилевский Э. Фонетико-акустическая база данных для многоязычного синтеза речи по тексту на славянских языках // Компьютерная лингвистика и интеллектуальные технологии: труды междунар. конф. Диалог'2006, Бекасово, 31 мая – 4 июня 2006 г./ Институт проблем информатики РАН; отв. ред.: Н.И. Лауфер [и др.]. – М.: Наука, 2006. – С. 357-363

[4] CMU US English Dictionary [Электронный ресурс] URL: https://github.com/cmusphinx/cmudict (дата обращения: 14.04.2022)

[5] Desklop/StressRNN: Modified version of RusStress (https://github.com/MashaPo/russtress) — python package for placing stress in Russian text using RNN (BiLSTM) and the "Grammatical Dictionary" by A. A. Zaliznyak (from http://odict.ru/). [Электронный ресурс] URL: https://github.com/Desklop/StressRNN

[6] dmort27/epitran: A tool for transcribing orthographic text as IPA (International Phonetic Alphabet) [Электронный ресурс] URL: https://github.com/dmort27/epitran
[7] explosion/spaCy: Industrial-strength Natural Language Processing (NLP) in Python [Электронный ресурс] URL: https://github.com/explosion/spaCy
[8] La'ncucki, Adrian. FastPitch: Parallel Text-to-speech with Pitch Prediction. ICASSP, 2021. – 5 p.
[9] Ponomareva Maria; Milintsevich Kirill; Chernyak Ekaterina; Starostin Anatoly (2017). Automated Word Stress Detection in Russian. Proceedings of the First Workshop on Subword and Character Level Models in NLP. Association for Computational Linguistics, 31−35. DOI: 10.18653/v1/W17-4104.
[10] RUSLAN: Russian Spoken Language Corpus For Speech Synthesis [Электронный ресурс] URL: https://ruslan-corpus.github.io/
[11] Zwicky A. M. On Clitics. – Bloomington: Indiana University Linguistics Club, 1977. – 41 p.