ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования

Национальный исследовательский университет
«Высшая школа экономики»

Факультет гуманитарных наук
Образовательная программа
«Фундаментальная и компьютерная лингвистика»

КУРСОВАЯ РАБОТА

На тему «Лингвистическая теория в распознавании речи»
Linguistic background in Speech Recognition

Студентка 3 курса
группы №192
Ревенко Дарья Сергеевна
Научный руководитель
Сериков Олег Алексеевич,
Приглашенный
преподаватель, Школа
лингвистики

Москва, 2021 г.

ВВЕДЕНИЕ

1.1. Определение переключения кода и его разновидности

В данный момент проблема распознавания переключения с одного языка на другой стоит довольно остро. Она заключается в том, что большинство моделей автоматического распознавания речи часто не могут справиться с наличием в одном предложении слов из разных языков, с тем, что фонемный состав слов в этих языках может сильно отличаться. Также одной из проблем является нехватка данных с переключением кода для многих языков.

Поскольку сейчас существуют языки, которые доминируют во всем мире и на которых говорят сотни миллионов людей, использование нескольких языков в разговоре уже не кажется чем-то странным. Из-за расширения влияния иностранных компаний и все бóльших межнациональных контактов, много людей в профессиональной деятельности начинают использовать второй язык. Обычно это один из 10 доминирующих языков.

Сейчас все чаще люди начинают употреблять кодовое переключение, особенно это заметно в сферах IT, маркетинга, телекоммуникаций и связанных с ними. Английский язык проник не только в рабочую сферу, им пользуются и в повседневной жизни. Большое количество людей использует интернет платформы именно на английском языке. Немалое количество научных работ от русских авторов публикуются и за рубежом, в большинстве школ этот язык изучается как первый иностранный. Кроме более технологичных сфер английский используется и в кулинарной, образовательной, коммерческой сферах. Таким образом большой пласт населения регулярно порождает кодсвитчинг или каким-то образом сталкивается с ним.

Помимо профессиональной надобности переключение кода может происходить и из-за разных социолингвистических, территориально-обусловленных или личностных факторов. Поэтому исследователи из разных сфер пытались определить, что такое переключение кода. Во многих работах термин переключение кода тесно связан с билингвизмом [Bloomfield, 1968; Haugen, 1972 и др]. Также часто утверждается, что существует несколько типов переключения кода – маркированное и немаркированное. К. Майерс-Скоттон [Myers-Scotton C., 1993] пишет: «Немаркированное переключение кодов имеет место тогда, когда говорящий следует установившемся в языковом сообществе правилам речевого поведения и переключается в соответствии с ожиданиями слушающего; маркированное переключение имеет место в том случае, если говорящий сознательно производит переключение таким образом, что это замечается собеседником как отклонение». Стоит также сказать, что в социолингвистике переключение кода – это скорее «переход говорящего в процессе речевого общения с одного языка (диалекта, стиля) на другой в зависимости от условий коммуникации».

Мы же будем опираться на то, что переключение кода – не только явление, присущее билингвальным людям, его могут употреблять и люди, не знающие второй язык на уровне носителя. Так, например, в среде русских программистов переключение кодов с русского на английский явление довольно частое [Burdygina M., 2020], при этом программист может не являться носителем английского или быть билингвом. Переключение кода мотивировано профессиональной деятельностью. Поэтому, в этом исследовании переключением кода будем называть попеременное использование элементов двух или более языков в рамках одного коммуникативного акта [Проценко Е.А., 2004].

Всего существует два вида кодового переключения: *интрасентенциальный* и *интерсентенциальный*. То есть переключение может происходить или внутри предложения, или между двумя предложениями. Пример интерсентенциального кодсвитчинга: *Делайте заголовки в два раза больше основного текста. Don't miss the boat*. Пример интрасентенциального кодсвитчинга: *Оптимизируем футбол с помощью Machine Learning.* В данной работе будут примеры как одного, так и другого вида. При распознавании речи межсентенциальное (то есть между предложениями) переключение кода можно найти, если определить границу монолингвальных предложений. Иногда найти границы предложений проблематично, но обычно это удается сделать при помощи фонетических и других критериев, поэтому такое кодовое переключение модели неплохо разбирают. При интрасентенциальном переключении не всегда удается определить момент, когда человек переходит на другой язык. Это может быть мотивировано психолингвистическими факторами (положением человека в обществе, статусом его собеседника) или, собственно, лингвистическими факторами (близость языков, количество и качество языковых контактов). Поэтому бóльший фокус будет сделан на выделении ошибок при распознавании интрасентенциального переключения кодов.

1.2. Модели для распознавания переключения кода

Существует несколько моделей для распознавания переключения кода, а также большое количество датасетов, которые поддерживают множество языков. С приходом новых технологий меняются методы, которыми пользуются люди для обучения машин распознаванию речи. Так, еще несколько лет назад для этой задачи использовался метод DNN-HMM [Ghoshal A., 2013], сейчас чаще используют End-to-End (E2E) подход. Довольно много исследований есть для Англо-Китайского и Китайско-Английского переключения кода [Shi X., 2019; Yu S. et al., 2003; Vu N. T. et al., 2012]. Существуют датасеты с большим количеством часов разговоров людей, которые время от времени переключаются с одного языка на другой для Англо-Китайского [Lyu D. C. et al., 2010], Хинди-Английского [Dey A., 2014], Испано-Английского [Deuchar, M., 2014]. Для нахождения в речи элементов с переключением кода предпринимались попытки использования нескольких моделей распознавания параллельно с идентификацией языка [Weiner J., 2012]. Также исследователи пытались прицельно находить именно данные с переключением кода. Это улучшило работу модели при

распознавании кодового переключения, но при этом повысило вероятность ошибки для одноязычного контекста.

Также часто из-за нехватки датасетов с переключением кода используется синтетическое создание таких наборов данных. Для этого берут данные из двух языков и поочередно вставляют фрагменты из одного языка в фрагменты из другого, чтобы получить и интра-, и интерсентенциальные предложения. Такой подход позволяет иметь большой набор данных, но он получается более ограниченным, чем в корпусе с переключением кода. Обычно переход с одного языка на другой происходит без нарушения синтаксических правил участвующих языков [Poplack S., 1980]. То есть качественно соединить два набора с разными языками может быть проблематично.

Стоит упомянуть, что из-за языковых контактов, территориальных факторов часто возникают ситуации, когда два языка сосуществуют, причем обычно один из них или миноритарный, или малоресурный. Так, например, есть датасеты для переключения кода с русского на коми и наоборот [Partanen N., 2018], а также для русско-якутского [Petukhova A., 2021] и русско-казахского [Mussakhojayeva S., 2022].

Интересно, что русский, хоть и является одним из 10 самых распространенных международных языков[1], не имеет пока датасетов для кодового переключения с русского на английский или наоборот. Но есть множество параллельных корпусов. Из них можно было бы составить модели для переключения, но это скорее всего повлияет на синтаксическую составляющую текста.

ПРЕДЫДУЩИЕ ИССЛЕДОВАНИЯ И ПОДХОДЫ К ДЕТЕКЦИИ ОШИБОК В МОДЕЛЯХ РАСПОЗНАВАНИЯ РЕЧИ

Rahhal Errattahi [Errattahi R., 2018] выделяет 3 типа ошибок при распознавании речи: слово может замениться на другое, оно может оказаться пропущенным при автоматической транскрипции, и обратное действие – новое слово может быть вставлено системой ASR. В то же время, [Salimbajevs A., 2015] пишет о других типах ошибок. Исследуются больше лингвистические единицы, чем структурные. Так, он говорит о важности расположения ошибки (в начале или конце слова), о длине слова (меньше 3 букв короткое), о правильности распознавания прошлого слова и о том, могло ли одно слово быть распознано как два.

[1] https://www.ethnologue.com/guides/ethnologue200

Стоит упомянуть другие исследования, посвященные выявлению и обработке ошибок в системах автоматического распознавания речи. Для этих целей используют в основном фреймворк для неавторегрессионного трансформера. Современные распознаватели речи (например, с E2E) содержат авторегрессионные модели (кроме Connectionist Temporal Classification), но авторегрессия не очень хорошо работает с такими “невидимыми” структурами, как переключение кода, поэтому неавторегрессионные методы для большей детекции ошибок могут помочь и в этом [Yizhou Peng, 2021]. Некоторые на основании существующих фреймворков и моделей создает автоматизированное распознавание ошибок на основе фонетики и морфологии.

Целью этого исследования является не только выявить ошибки в моделях распознавания речи с переключением кода, но и попытаться их сгруппировать для последующей автоматизации.

Для достижения поставленных целей необходимо решить следующие задачи:

1. изучить и проанализировать литературу по теме исследования;
2. собрать автоматизированные данные с переключением кода;
3. применить модель для распознавания речи;
4. обработать данные отдельно для русского и английского, также отдельно для слов и предложений;
5. оценить, насколько хорошо произошло распознавание, какие есть ошибки;
6. применить коррекцию ошибок и сравнить с изначально распознанным текстом;
7. оценить полученные результаты;

Для оценки результата были выбраны метрики Character Error Rate (CER) и Word Error Rate (WER). Эти метрики являются традиционными для оценки системы распознавания речи [Shi X., 2019; Ali A. et al, 2021; Yu D., 2016].

ДАННЫЕ И МАТЕРИАЛЫ

Поскольку данные для интерсентенциального кодового переключения довольно просто найти или составить самим из имеющихся корпусов, мы решили пойти немного другим путем. Модели для распознавания речи с переключением между предложениями справляется довольно неплохо [Mussakhojayeva S., 2021], в то время как внутри предложения переключение распознать все еще довольно проблематично.

Для исследования были выбраны данные из сферы IT и дизайна. Именно в этой нише встречается самое большое количество заимствований и переходов с одного языка на другой, преимущественно английский. В данной работе рассмотрены случаи в основном интрасентенциального переключения, его системы пока распознают хуже, чем переключение между предложениями. Поскольку данных для русско-английского переключения нет, а собирать их вручную трудоемко, было решено создать их самим.

3.1. Процесс сбора данных

Изначально были получены все статьи из нескольких сотен (чуть больше 300) страниц сайта Habr[2]. Этот сайт был выбран, поскольку на нем есть статьи с большим содержанием иностранных слов. Также можно заметить, что в разных статьях довольно часто используются одни и те же английские слова, что поможет натренировать модель и покажет, насколько хорошо одинаковые слова могут быть распознаны.

После парсинга каждая статья проверяется на наличие в ней английских слов (статьей считался текст, где было больше 20 слов). Иногда на Habr попадаются английские статьи, поэтому критерием отбора было наличие не менее 15 % английских слов в статье, но и не более 90 %. Всего отобрано немного больше 50 веб-страниц, в общем счете 3555 предложений. В конце предложения записываются в файл.

Далее при помощи Yandex Speechkit[3] для каждого предложения из файла синтезируется речь. Каждый файл возвращается с расширением .raw, а затем конвертируется в .wav файл с частотой дискретизации 48 кГц. Запись в формате lpcm. Поскольку исследуется русско-английское переключение кодов, то и язык для синтеза был выбран русский. Было решено оценить работу программы на 1000 предложениях. На выходе получается папка с 1000 аудиофайлами, общая длительность всех аудио составляет 8120,882 секунды или 2,256 часа.

МЕТОДЫ

На основании сгенерированных данных производится анализ ошибок распознавания речи на участках переключения кодов. После сбора данных и синтеза каждого предложения из этого набора в речь происходит процесс распознавания речи при помощи wav2vec2-large-xlsr-53-russian модели[4], которая является развитием

[2] https://habr.com/
[3] https://cloud.yandex.ru/docs/speechkit/
[4] https://huggingface.co/jonatasgrosman/wav2vec2-large-xlsr-53-russian

архитектуры wav2vec. После распознавания 1000 файлов создается файл для последующего выравнивания. Выравнивание производится при помощи алгоритма fast align[5] [Chris Dyer, 2013]. После преобразований получаем список предложений, в котором каждое слово из распознанного предложения сопоставляется со словом из исходного. После этого были посчитаны Character Error Rate (CER) и Word Error Rate (WER) для всех пар предложений, а также отдельно для русских и английских слов. Далее автоматически выделяем определенные закономерности, которые по нашей гипотезе могли бы улучшить CER или WER. В конце создаем несколько датафреймов для последующего анализа.

Остановимся подробнее на каждом из шагов. Модель для русского была выбрана, поскольку интересно было посмотреть на те ошибки, которые она допускает при распознавании русского с добавлением иногда английских слов или даже вставкой английских предложений. Важными критериями стали результаты оценки модели на данных Common Voice[6]. Выбранная модель[7] одна из самых новых, и на приведенных данных показывает хорошие CER и WER. Также она использует трансформеры, на данном этапе такие модели лучше себя показывают.

4.1. Первый этап работы, 1-to-1

После распознавания создается файл определенного формата. Пример строки из созданного файла:

навагейшендрор боковая панель навигации ||| Navigation Drawer — боковая панель навигации.

Слева – распознанный текст, справа – исходный.

Это сделано для того, чтобы далее воспользоваться fast align. Также на этом этапе выполняется препроцессинг текста. Распознанный текст состоит только из букв и пробелов, поэтому в исходном тексте нужно избавиться от всей пунктуации, заглавные буквы превратить в строчные, буквы “ё” заменить на “е”, убрать все символы, не относящиеся к буквам и цифрам. Далее приступаем к выравниванию. На выходе после выравнивания мы получаем файл, состоящий из набора пар i-j для всех предложений, каждая из пар i-j указывает, что i-е слово (начиная с нулевого индекса) левого языка выровнено по j-му слову правого предложения. Для дальнейшей работы и анализа

[5] https://github.com/clab/fast_align
[6] https://commonvoice.mozilla.org/ru/datasets
[7] https://huggingface.co/jonatasgrosman/wav2vec2-large-xlsr-53-russian

нужно иметь для слова из распознанного предложения его пару из исходного. Поскольку выравнивание - это индексы в двух списках, найти слова по этим индексам оказалось не так сложно. Ниже приведен пример обработки сырого текста для выравнивания и последующей работы:

(1) Navigation drawer боковая панель навигации. (текст для предобработки)
(2) навагейшендрор боковая панель навигации (распознанный текст)
(3) navigation drawer боковая панель навигации (текст после препроцессинга)
(4) 0-0 0-1 1-2 2-3 3-4 (alignment, первое слово из распознанного перешло в первое и второе слово из исходного)
(5) навагейшендрор навагейшендрор боковая панель навигации (распознанный текст после конвертации, исходный остается таким же, как был)

4.2. Второй этап работы, many-to-1

Можно заметить, что в предложении из примера одно слово кодирует сразу два. Это часто встречается в нашем корпусе, что позволяет выдвинуть гипотезу: если после обычного выравнивания еще раз пройтись по индексам P (распознанного текста), найти одинаковые, сопоставить их с индексами T (исходного текста), после чего таким же образом сопоставить сами слова с индексами, то CER станет меньше. После автоматической обработки таким образом для предыдущего примера получается такая структура:

(6) навагейшендрор боковая панель навигации (распознанный текст после добавления обработки)
(7) navigationdrawer боковая панель навигации (исходный текст после добавления обработки)

Будем называть данные, в которых несколько слов из распознанного текста перешли в одно слово из исходного текста данными many-to-1. Напротив, данные из прошлого этапа работы будут называться 1-to-1, поскольку в этом случае мы считаем, что одному слову из распознанного соответствует одно слово из исходного.

4.3. Третий этап работы, МФА

Сама по себе такая обработка никак не увеличит CER, поскольку эта метрика означает частоту ошибок по символам, а когда символы из разных языков, такая эвристика не поможет. Для правильной работы CER нужно иметь единую систему символов. Поэтому следующим шагом является преобразование всех слов в Международный фонетический алфавит (МФА). Такое преобразование позволяет

понять, насколько хорошо модель распознавания справляется с английскими словами, а также увидеть разные виды ошибок и то, с чем они связаны. Преобразование в МФА проводится отдельно для русских и английских слов. Используются соответствующие модели из библиотеки epitran[8] для преобразования слов русского и английского языков.

Далее мы в основном будем работать не с предложениями, а со словами, поскольку наш интерес именно в том, почему существуют ошибки в словах при переключении кода. Для данных из первого этапа работы каждое слово из распознанного текста проходит через русскую модель, каждое английское через английскую. С данными из второго этапа все немного сложнее. Например, в распознанном тексте есть слово “ванеджина”, которое в исходном тексте соответствует двум словам “в unigine”, одно из которых русское, другое английское. Мы не хотим терять эту сущность, поэтому для таких случаев отделяем все русское от английского, отдельно прогоняем русскую сущность через модель для русского, английскую - для английского.

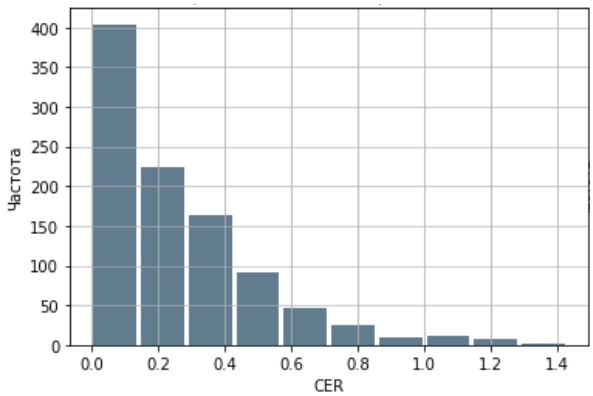
В конце для всех пар предложений считаем CER и WER, также CER считаем для каждого английского и русского слова из всего текста по отдельности для 1-to-1 и many-to-1 данных соответственно. Стоит также упомянуть, что обычная метрика CER, которая использовалась до этого, с данными МФА работает не всегда хорошо (для *botom* и *batəm* CER = 40%, хотя мы понимаем, что это одно и то же слово), поэтому мы используем библиотеку abydos[9] и один из ее алгоритмов Phonetic Edit Distance[10]. С его помощью можно посчитать расстояние Левенштейна между строками в МФА, алгоритм сравнивает фонемы на основе их внешнего сходства. Используемый метод возвращает нормализованное фонетическое расстояние редактирования между двумя строками.

В итоге получаем несколько датафреймов, отражающих основные характеристики полученных текстов. Их мы и будем анализировать для определения ошибок и качества модели на английских словах.

[8] https://github.com/dmort27/epitran
[9] https://abydos.readthedocs.io/en/latest/index.html
[10] https://abydos.readthedocs.io/en/latest/abydos.distance.html#abydos.distance.PhoneticEditDistance

РЕЗУЛЬТАТЫ

5.1. Оценка распознавания предложений



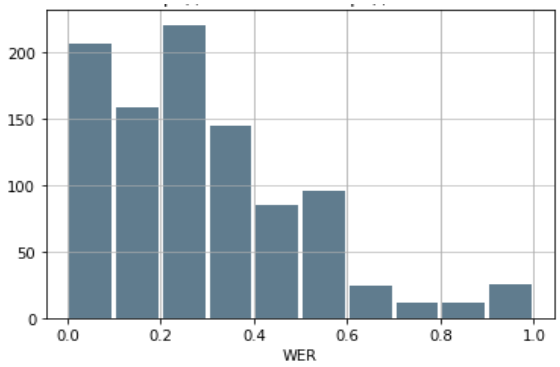


График 1. Распределение CER в предложениях

График 2. Распределение WER в предложениях

Из графика 2 видно, что для предложений с переключением кода WER в основном меньше 40%. Это не самое хорошее значение, но для данных с кодсвитчингом среднее значение обычно чуть больше 20% [Mussakhojayeva S., 2021]. Среднее значение WER для всех наших предложений - 28,14%. Без применения наших эвристик (конвертации всех слов в МФА и объединения одинаковых для распознанного текста слов) английский текст и русский все еще не могут хорошо вместе распознаться, значит нужно посмотреть на эти метрики с эвристиками.

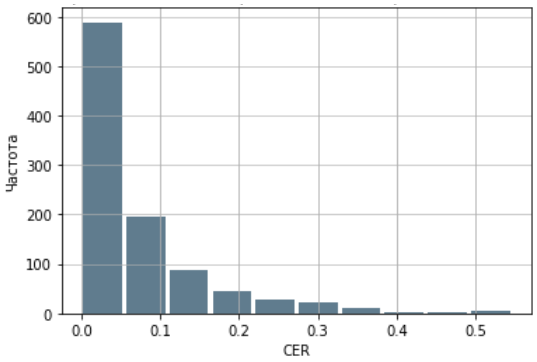


График 3. Распределение CER в предложениях, переведенных в МФА

Сравнивая график 1 и график 3, мы приходим к выводу, что CER намного лучше: теперь больше 550 предложений распознаются с CER меньше 5%. Те, что распознаются с высоким значением CER можно отнести к нескольким категориям: в исходном тексте много чисел, поэтому распознавание работает значительно хуже; часть слов не распозналась; или в предложении почти все слова английские. Всего таких предложений оказалось 27, это меньше 3% от всего числа предложений.

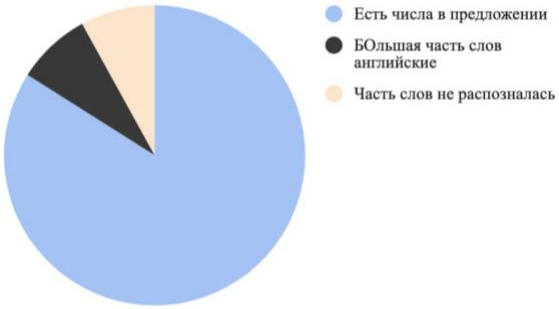


График 4. Ошибки в предложениях с CER больше 3

На графике 4 видно распределение по каждому из типов ошибок. Чаще всего грубые ошибки распознавания встречаются в предложениях, где есть числа.

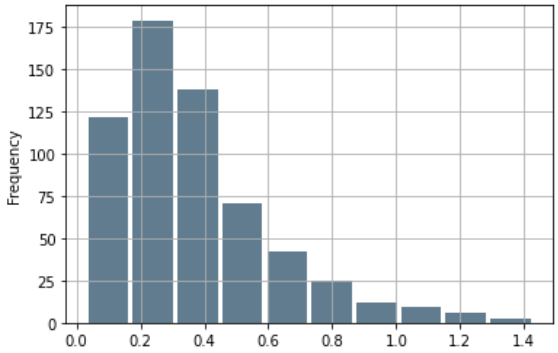


График 5. Распределение CER только в предложениях с английскими словами

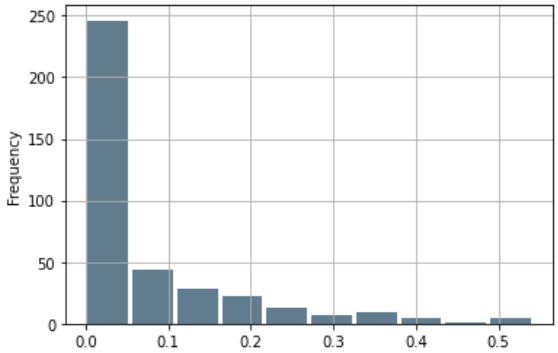


График 6. Распределение CER только в предложениях с русскими словами, без английских

Из графиков 5 и 6 видно, что предложения с русскими словами распознаются намного лучше, чем с английскими. CER меньше 5% для большей части русских предложений и меньше 20% для большинства предложений с английскими словами. Отсюда можно сделать вывод, что сосуществование русских слов рядом с английскими нивелирует высокие значения ошибок в словах для предложений. Подтвердим это также графиками, которые показывают CER для всех русских слов.

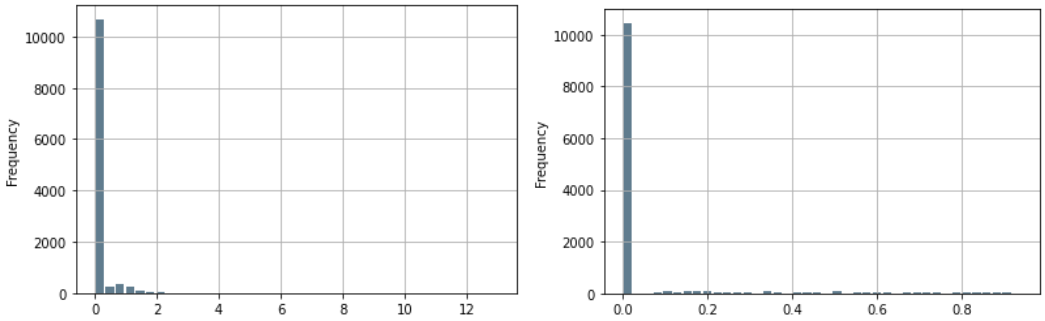


График 5. CER для русских слов в МФА. Слева - CER для графем, справа - CER для фонем

Можно заметить, что абсолютное большинство русских слов распознаются правильно, особенно радует график справа. Есть незначительные проблемы там, где слова распознаются с высоким CER, но из общей выборки они составляют меньше 5%. Это подтверждает идею о том, что CER при МФА получается достаточно низким, даже если в предложении есть английские слова.

5.2. Сравнение CER для английских данных 1-to-1 и many-to-1, слова в МФА

Перейдем к сравнению данных и результатов 1-to-1 экспериментов с теми, для которых мы взяли дополнительные параметры (many-to-1).

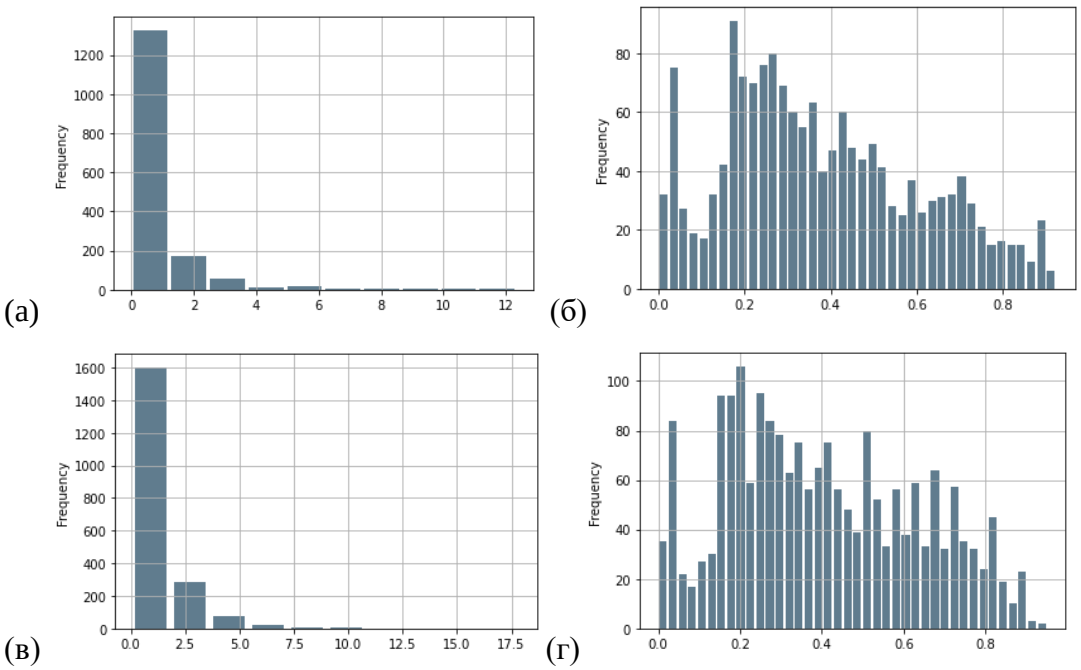


График 7. Распределения CER по английским словам в МФА (а, б - данные с нововведениями(many-to-1), в,г - данные просто с выравниванием (1-to-1); а,в - обычный CER, б,г - CER от abydos, специально для МФА)

Итак, сравним графики с CER для графем и CER специально для слов в формате МФА. Сразу можем заметить, что CER для графем (а и в) показывает результат

намного хуже, чем CER для МФА. В нем есть показатели больше 1, поскольку он не нормализован. Также показателей больше 1 не так мало. Стоит отметить, что в данных с эвристиками даже CER для графем чуть лучше, чем для данных 1-to-1. Эти графики все же не так примечательны, как графики (б) и (г). На них уже лучше видно, как распределяются ошибки. Есть общая тенденция: во-первых, из-за МФА распределение получается более однородное; во-вторых, видно, что у many-to-1 данных в распределении почти нет больших скачков, все стремится к спаду по мере движения к более высоким значениям CER, в то время как у графика (г) выбросы начинаются как раз, когда CER > 50%. Это может означать, что при существовании общей тенденции к уменьшению количества данных при увеличении значения CER, есть слова, которые не дают данным из 1-to-1 делать это плавно, без выбросов. Если обратимся к начальным значениям, которые меньше 20%, то заметим, что здесь ситуация более непонятная. В данных 1-to-1 больше таких значений, чем в данных с эвристиками.

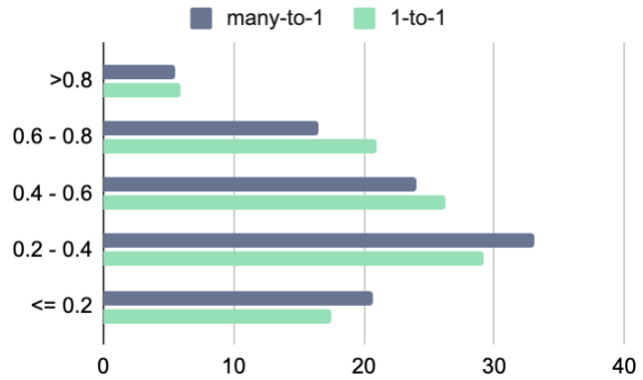


График 8. Процентное соотношение CER для данных с нововведениями и без

На графике 8 четко видно, что по относительным показателям many-to-1 обгоняет 1-to-1 на более низких значениях CER. Даже в том месте, где CER <= 20%, все равно модель с эвристиками отработала лучше. При этом показатели плохого Character Error Rate не сильно далеки друг от друга. Все же данные many-to-1 показывают себя лучше в поведении с CER, чем данные 1-to-1.

5.3. Группировка ошибок и признаки повышения CER

Далее рассмотрим данные, для которых CER показал лучший результат, то есть данные с эвристиками (конвертация в МФА, постановка в соответствие одному значению из распознанного предложения все слова, к которым оно относится в

истинном предложении, использование метрики CER, специально для слов с МФА). Посмотрим на то, какого типа ошибки чаще всего встречаются.

Начнем с ошибок с самым большим CER (>80%). При помощи тщательной интроспекции удалось выяснить, что ошибки в таких словах делятся на несколько типов. Зная о возможности встречи таких слов, мы можем улучшить качество распознавания. На графике 9 показано процентное соотношение ошибок в зависимости от их вида для CER > 80%.

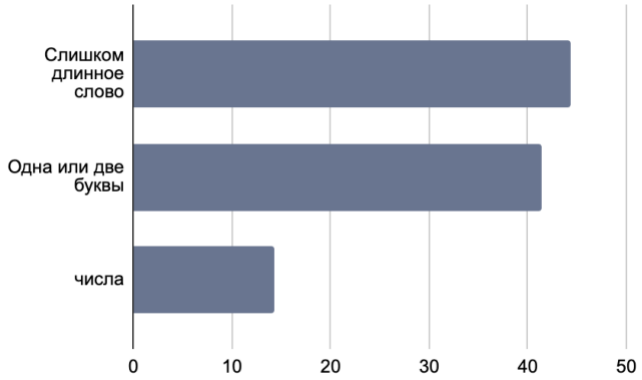


График 9. Процентное соотношение ошибок при CER > 80%

Можно заметить, что для данных many-to-1 хоть и уменьшился CER, наш метод оказался не таким универсальным, как хотелось бы. Получившиеся группы ошибок:

1) Ошибки в очень длинных словах
2) Ошибки, связанные с тем, что слово из истинного текста распозналось или выровнялось как буква
3) Ошибки в данных с английскими буквами вместе с числительными

Слишком длинные слова - это в основном те, которые неправильно выровнялись, или те, для которых истинное слово оказалось еще длиннее (из-за того, что мы конкатенировали те строки, в которых истинные слова относятся к одному и тому же предсказанному). Соответственно, предсказанное слово оказывается частью истинного. Также, например, если предложение слишком длинное с большим количеством английских слов, или если какое-то слово оказывается опущенным в процессе выравнивания, то скорее оно не распознается правильно. Пример: “паи” из предсказанного текста должно стать “ui”, но не распознается и вообще уходит из итоговой выборки, вместо него в соответствие ui ставится следующее слово “компонентам”.

В тех местах, где предсказанные слова состоят из одной или двух букв, истинные обычно длиннее. Скорее всего, или выравнивание срабатывает не так, как

нам хотелось бы, или во время синтеза речи эти слова произносятся неправильно. Пример: “randomize”, стоящее в конце предложения распознается как просто “p”, соответственно и значение для CER у него высокое.

Оставшаяся группа - слова с числительными. Поскольку в исследовании не делается предобработка числительных, в истинных данных они остаются числами, а в распознанных становятся словами. Поскольку с числами иногда стоят латинские буквы (скорее из-за специфики выбранного домена для исследования), они попадают в выборку. Таких слов не очень много, их можно просто убирать.

Далее рассмотрим слова, у которых CER от 60% до 80%, поскольку это все равно довольно высокий показатель. Здесь к уже обозначенным типам ошибок добавляются ошибки распознавания части слова как целого (“dynamic” распознается как “дай”, часть слова потерялась), а также распознавания русского слова как иностранного. Поскольку при склеивании мы не смотрим на количество букв в русском и иностранном слове, которые конкатенируем, мы не можем идентифицировать такие вещи. В остальных местах ошибки такие же, просто из-за большей схожести с истинным словом они попали в другой кластер и имеют меньшую оценку CER.

Помимо ошибок стоит упомянуть длину предложений. Во многом, слова начинают хуже распознаваться, когда предложение слишком длинное, при этом в нем много английских слов. Иногда количество слов в распознанном тексте превышает количество слов в истинном в 1,5 раза. Думаю, такие предложения стоит убирать еще на этапе препроцессинга.

5.4. Сравнение самых хорошо распознанных слов в 1-to-1 и many-to-1

Следующая ступень поиска ошибок – между данными, в которых используются разные методы. Здесь CER для данных many-to-1 довольно низкий. Рассмотрим все слова, относящиеся к 1-to-1 с CER <= 20%. То же самое сделаем для many-to-1. Мы это делаем для того, чтобы найти, чем различаются слова для самых лучших значений CER. Оказывается, что в данных 1-to-1 есть 15 слов, у которых CER <= 20% и которых нет в many-to-1. Это означает, что на этих позициях данным из набора с нашими эвристиками что-то помешало, и их CER повысился. После рассмотрения всех 15 слов становится понятно, что такие ошибки в данных связаны как раз с нововведением. Так, например, в 1-to-1 слово *outlined*, МФА для которого *awtlajnd* распозналось как *aut-l*ʲ*ejnd*, CER меньше 20%. Но в many-to-1 это слово сконкатенировалось с другим и

получился *awtlajnds*. CER уже становится больше 20%. Это говорит о том, что не всегда хорошо справляется наше нововведение, хоть оно все еще и обыгрывает обычный alignment в качестве. Обратим внимание также на ошибки, которые, наоборот, должны были быть сделаны. Так, например, *ta* распознается как *ʃi*. При этом CER оказывается меньше 20%. Новая модель соединяет строки, поэтому значение CER получается большим, что правильно. Также есть случаи, когда одному истинному слову соответствует несколько распознанных (обратно к тому, что мы делали). Например, сконкатенированный истинный *ænəmejʃənzimport* относится и к *import*, и к *majʂins*. Скорее всего, если бы распознанные слова тоже соединялись, CER стал бы меньше, и, возможно, помог бы также избавиться от нескольких неправильных слов там, где CER > 80%.

5.5. Ошибки при интерсентенциальном переключении кода

Отдельно стоит рассмотреть ошибки в предложениях, которые полностью на английском языке и стоят между другими русскими предложениями. Для таких мы тоже все слова переводятся в МФА, и считается CER специально для слов в МФА). Средний CER для таких предложений оказывается 24%. То есть даже английские слова нормально распознаются. Стоит отметить, что хуже всего распознаются предложения, где помимо английских слов есть цифры, лучше всего те, в которых мало слов. Из всего датасета нашлось 14 предложений на английском, самое длинное из которых имеет длину 12 слов. Это маленький набор данных. Такая выборка говорит о том, что в обычной жизни люди чаще используют переключение на английский не между предложениями, а между словами в одном предложении. CER почти везде получается в пределах нормы для распознавания кодсвитчинга.

5.6. Сравнение букв в словах

Наконец, мы решили посмотреть на самые распространенные звуки в распознанных и истинных словах для разного процента ошибок. Рассматриваются звуки для начала слова, гласные, а также согласные, которые встречаются чаще всего. Это позволяет понять, насколько фонетические характеристики звука зависят от того, насколько хорошо он распознается в слове.

	T_less_7	T_more_80	P_less_7	P_more_80
Первая буква слова	n,m,æ	i,s,ə	n,m,i	i,v,t
Гласная	ə,o,a	ə,i,ɛ	a,o,e	a,e,i
Согласная	n,j,t	n,m,t	n,t,s	t,n,s

Таблица 1. Распределение букв в словах в зависимости от процента CER

Расшифровка названий столбцов: T_less_7 – истинное слово, у которого значение CER < 7%; T_more_80 – то же самое, но значение CER > 80%. Остальные два столбца для предсказанных слов. Можно заметить, что первый звук в словах, которые распознаются с CER > 80% – гласная. Возможно, это зависит от количества данных, и на других звуки будут отличаться, но все же можно предположить, что когда слово начинается с гласной, скорее оно хуже будет распознаваться.

ЗАКЛЮЧЕНИЕ

Переключение кода встречается все чаще, но системы распознавания так и не научились его качественно находить. В работе предпринята попытка группировки ошибок, также предложен метод для улучшения распознавания после выравнивания. Выявлены основные группы ошибок:

1) Ошибки в очень длинных словах
2) Ошибки, связанные с тем, что слово из истинного текста распозналось или выровнялось как буква
3) Ошибки в данных с английскими буквами вместе с числительными
4) Ошибки в словах, где распознанное значение является частью истинного
5) Ошибки, где одному истинному слову соответствует несколько распознанных

Предложен алгоритм исправления ошибок на участках с переключением кода (это конвертация всех слов в МФА; если одно распознанное слово ссылается на несколько истинных, то объединение истинных слов в одно), с его помощью удается снизить показатель CER, а значит наша гипотеза частично верна. Нашлись случаи, когда наш алгоритм наоборот повышал значение CER. Такие вещи можно разбирать мануально или при помощи поиска оптимального варианта из двух. Если у нас есть одинаковые для распознавания слова в одном и том же предложении на одинаковых позициях, при этом у них разные истинные значения, можно находить CER обоих и принимать за правильное то значение, где CER меньше. Также в качестве дальнейшей работы можно использовать данные не только распознанных значений, но и истинных.

Так, если у нас для разных распознанных слов одинаковое истинное слово, то скорее всего CER покажет лучший результат, если мы объединим соответствующие распознанные слова в одно. Также внедрение МФА очень сильно повысило качество моделей. Из распространенных ошибок, которые мы смогли найти, были те, на которые обращали внимание и другие исследователи. Думаю, такие ошибки также можно автоматизированно выявлять, что еще больше улучшит работу модели.

В целом, нехватка данных также сказывается на работе модели. Интересно было бы выяснить, насколько модель хорошо справляется с несинтезированной речью с переключением кодов.

Наш эксперимент показывает, что ошибки можно сгруппировать и в дальнейшем автоматически обработать для улучшения показателей CER и WER. Нужно обращать внимание не только на количественные, но и на лингвистические параметры, а также пытаться автоматизировать распознавание таких вещей.

СПИСОК ЛИТЕРАТУРЫ

1. A. A. Petukhova, E. O. Sokur. 2021. Yakut-Russian Corpus of Code-Switching, Moscow: International Laboratory of Language Convergence, Higher School of Economics. (Available online at: http://lingconlab.ru/cs_yakut, accessed on 27.05.2022.)
2. Ali A. et al. Arabic code-switching speech recognition using monolingual data //arXiv preprint arXiv:2107.01573. – 2021.
3. Bloomfield L. Language. – Motilal Banarsidass Publ., 1994.
4. Burdygina M. Code-Switching (Russian-English) In The Discourse Of It-Specialists From Moscow //Higher School of Economics Research Paper No. WP BRP. – 2020. – Т. 103.
5. Deuchar M. et al. 5. building bilingual corpora //Advances in the Study of Bilingualism. – Multilingual Matters, 2014. – С. 93-110.
6. Dey A., Fung P. A Hindi-English Code-Switching Corpus //LREC. – 2014. – С. 2410-2413.
7. Dutta S. et al. Error correction in asr using sequence-to-sequence models //arXiv preprint arXiv:2202.01157. – 2022.
8. Dyer C., Chahuneau V., Smith N. A. A simple, fast, and effective reparameterization of ibm model 2 //Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. – 2013. – С. 644-648.
9. Errattahi R., El Hannani A., Ouahmane H. Automatic speech recognition errors detection and correction: A review //Procedia Computer Science. – 2018. – Т. 128. – С. 32-37.
10. Ghoshal A., Swietojanski P., Renals S. Multilingual training of deep neural networks //2013 IEEE international conference on acoustics, speech and signal processing. – IEEE, 2013. – С. 7319-7323.
11. Haugen E. The American Dialects of Norwegian // The Norwegian Language in America: A Study in Bilingual Behavior. Vol. 1. Copyright Date, 1953. P. 1-317.
12. Lyu D. C. et al. Language identification in code-switching speech using word-based lexical model //2010 7th International Symposium on Chinese Spoken Language Processing. – IEEE, 2010. – С. 460-464.
13. Matarneh R. et al. Speech recognition systems: A comparative review. – 2017.
14. Mussakhojayeva S., Khassanov Y., Atakan Varol H. A study of multilingual end-to-end speech recognition for Kazakh, Russian, and English //International Conference on Speech and Computer. – Springer, Cham, 2021. – С. 448-459.
15. Myers-Scotton C. Duelling Languages: Grammatical Structure in codeswitching. Oxford: Clarendon Press, 1993. 285 p.
16. Myers-Scotton C. Multiple voices: an introduction to bilingualism. Blackwell publishing. 2006. 472 p.

17. Partanen N. et al. Dependency parsing of code-switching data with cross-lingual feature representations //Proceedings of the Fourth International Workshop on Computatinal Linguistics of Uralic Languages. – 2018. – С. 1-17.
18. Peng Y. et al. Minimum word error training for non-autoregressive Transformer-based code-switching ASR //ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). – IEEE, 2022. – С. 7807-7811.
19. Poplack S. Sometimes i'll start a sentence in spanish y termino en espanol: toward a typology of code-switching1. – 1980.
20. Riehl C. M. Code-switching in bilinguals: Impacts of mental processes and language awareness //Proceedings of the Fourth International Symposium on Bilingualism. – Somerville, MA : Cascadilla Press, 2005. – С. 1945-1960.
21. Salimbajevs A., Strigins J. Latvian speech-to-text transcription service //Sixteenth Annual Conference of the International Speech Communication Association. – 2015.
22. Shi X., Feng Q., Xie L. The asru 2019 mandarin-english code-switching speech recognition challenge: Open datasets, tracks, methods and results //arXiv preprint arXiv:2007.05916. – 2020.
23. Sichyova O. N. A note on Russian–English code switching //World Englishes. – 2005. – Т. 24. – №. 4. – С. 487-494.
24. Sitaram S. et al. A survey of code-switched speech and language processing //arXiv preprint arXiv:1904.00784. – 2019.
25. Vu N. T. et al. A first speech recognition system for Mandarin-English code-switch conversational speech //2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). – IEEE, 2012. – С. 4889-4892.
26. Weiner J. et al. Integration of language identification into a recognition system for spoken conversations containing code-switches //Spoken Language Technologies for Under-Resourced Languages. – 2012.
27. Yu D., Deng L. Automatic speech recognition. – Berlin : Springer, 2016. – Т. 1.
28. Yu S. et al. Chinese-English bilingual speech recognition //International Conference on Natural Language Processing and Knowledge Engineering, 2003. Proceedings. 2003. – IEEE, 2003. – С. 603-609.
29. Проценко Е. А. Проблема переключения кодов в зарубежной лингвистике (краткий обзор литературы за последние десятилетия) //Вестник Воронежского государственного университета. Серия: Лингвистика и межкультурная коммуникация. – 2004. – №. 1. – С. 123-127.