ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

**Факультет Санкт-Петербургская школа
физико-математических и компьютерных наук**

Жемчужина Елизавета Всеволодовна

Семантически обусловленные методы токенизации текстов

Выпускная квалификационная работа - БАКАЛАВРСКАЯ РАБОТА
по направлению подготовки *01.03.02 Прикладная математика и информатика*
образовательная программа «Прикладная математика и информатика»

Рецензент
кандидат физико-математических наук, ООО "Яндекс", руководитель группы исследования краудсорсинга Яндекс.Толока
_____
Д. А. Усталов

Руководитель
доктор физико-математических наук, профессор, департамент информатики
_____
Б. А. Новиков

Санкт-Петербург 2023

Содержание

Аннотация

Методологии выделения крупных текстовых единиц и небольших, активно используемых языковыми моделями, значимо отличаются. Операционные возможности моделей с разноуровневыми единицами текста не до конца исследованы. При этом не существует единой теории масштабирования дискретизации текста. Основной целью данной работы является создание нового глобального подхода к оценке единиц дискретизации языка, на пересечении статистики, лингвистики и глубокого обучения. В ходе работы был создан метод дискурсивной сегментации текстов, который не требует дообучения и может использоваться для верхнеуровневой дискретизации текста. Обширные эксперименты с использованием разных алгоритмов токенизации и размеров словаря выявили условия декомпозиции распределения на два, характеризующиеся разноплановыми статистическими и семантическими свойствами в рамках когерентных групп токенов. На основе этих экспериментов, а также дополнительной экспертной лингвистической оценки была предложена новая классификация текстовых единиц. Была разработана методология автоматического определения типов текстовых единиц, использующая архитектуру большой языковой модели. С помощью этого фреймворка был проведена в рамках заявленной типологии классификация активного словаря модели.

Ключевые слова: обработка естественного языка, закон Ципфа, токенизация, дискурсивная сегментация

The methodologies for identifying large-scale textual units and smaller ones actively used by language models significantly differ. The operational capabilities of models with different levels of text units are not fully explored, and there is no unified theory for scaling the discretization of text. The main objective of this study is to develop a novel global approach to evaluate language discretization units at the intersection of statistics, linguistics, and deep learning. In this work, a method for discourse segmentation of texts was created, which does not require fine-tuning and can be used for high-level text discretization. Extensive experiments using various tokenization algorithms and vocabulary sizes revealed conditions for decomposing the distribution into two coherent token groups characterized by diverse statistical and semantic properties. Based on these experiments and additional expert linguistic assessment, a new classification of text units was proposed. A methodology for automatically determining the types of text units was developed, using a large-scale language model architecture. Using this framework, the active vocabulary of the model was classified according to the proposed typology.

Keywords: natural language processing, Zipf's law, tokenization, discourse segmentation

Введение

Различные современные модели обработки естественного языка (NLP) разрабатываются и обучаются на основе гипотезы дистрибутивной семантики. А именно, что лингвистические элементы, используемые в схожих контекстах, имеют схожие значения [17]. В математике самоподобный объект точно или приблизительно похож на часть самого себя (т.е. целое имеет ту же форму, что и одна или несколько частей). Многие объекты в реальном мире, такие как береговые линии, статистически самоподобны: их части демонстрируют одинаковые статистические свойства во многих масштабах; наиболее известным объектом, обладающим самоподобием является фрактальная структура. Хотя в таком строгом смысле в естественном языке не наблюдается полного самоподобия, определенные его элементы все же присутствуют: буквы складываются в слова, слова в предложения, предложения в абзацы, абзацы в страницы текста и т.п. Тем не менее область того, как большие нейросетевые архитектуры для обработки естественного языка, такие как BERT, GPT-3, XLNet, DeBERTa, PaLM [15, 3, 50, 18, 12], учитывают различные внутренние характеристики этих текстовых единиц и их качественные изменения при смене уровня дискретизации, не до конца исследована. Таким образом, остается открытым ряд вопросов: Как соотносится дистрибуционная гипотеза и различные уровневые единицы языка? Дискретизация текстов на какие смысловые блоки наиболее эффективна с точки зрения обработки языковой моделью? Как оценить качественное наполнение словаря языковой модели? В фокусе данного исследования находится изучение различных методологий дискретизации текста на разных уровнях языка, выработка собственной типологии лингвистических единиц, использующей как статистические, так и

семантические свойства и ее применение к словарям широко используемых языковых моделей.

Существует ряд исследований[20, 23, 48] по разделению текста на крупные смысловые единицы: выделению элементарных дискурсивных единиц и установлению отношений между ними. В связи с указанным выше сохраняющемся самоподобием на верхнем уровне организации текста, представляется важным не упустить из рассмотрения выделение подобных крупных комплексных единиц естественного языка. Кроме того, упомянутые работы представляют крайне сложные и перегруженные архитектуры. Мы в то же время опираемся на следующую гипотезу: крупная языковая модель без серьезного дообучения, требующего тысячи часов вычислений на мощных графических процессорах, способна выделять указанные смысловые куски с высокими показателями метрик.

Также существуют более классические методы разделения текста на токены. В то же время, закон Ципфа гласит, что при наличии некоторого корпуса высказываний на естественном языке, частота встречаемости любого слова обратно пропорциональна его рангу в таблице частот Zipf (1932) [52]. По мере того как размер словаря (или объем данных) растет, это распределение ранг-частота дает "тяжелый хвост" - значительное количество лингвистических элементов, частота которых не является достаточной для того, чтобы дистрибуционная семантика показывала себя эффективно. Большинство современных методов NLP отсекают этот "хвост", введя понятие "токена", и ограничивают количество единиц в словаре модели. Однако тот факт, что модели, основанные на дистрибуционной гипотезе, обучаются на корпусах, которые следует закону Ципфа, поднимает ряд интересных вопросов, связанных с семантической составляющей и возможностью группировки токенов в зависимости от соответствия определенным параметрам закона Ципфа. Это определяет фокус при работе с методами дискретизации текста на

небольшие смысловые единицы. Также, подходя всесторонне к исследуемому предмету, мы обращаемся помимо всего прочего к теоретико информационным оценкам языковых единиц.

Важно подчеркнуть, что способы, предлагаемые для выделения крупных единиц текста, принципиально отличаются от алгоритмов токенизации. Это в свою очередь влечет отсутствие перехода от одного масштаба дискретизации текста к другому и не позволяет в рамках единой системы изучить то, как языковые модели воспринимают разноуровневые единицы текста. В данном исследовании мы сначала рассматриваем методологии выделения крупных кусков, а затем обращаемся к коротким и предлагаем подход, связывающий эти единицы масштабирования текста в единую систему.

Определение ключевых терминов

Токен - лингвистическая единица, состоящая из последовательности символов, подслов или целых слов

Гранулярность последовательности - характерный масштаб элементов, на которые последовательность разделяется при препроцессинге или "алфавит" в теоретико-информационном смысле

Элементарная единица дискурса (EDU) - минимальная законченная смысловая единица текста, связываемая с другими единицами с помощью дискурсивной связи

Дискурсивная сегментация - разделение текста на EDU

Цель и задачи

Целью данной работы является исследование различных уровней дискретизации текста с использованием больших языковых моделей и выработка семантически ориентированного подхода к оценке единиц гранулярности языка на основании закона Ципфа.

Задачи:

- Предложить способ извлечения крупных единиц дискретизации
- Выработать гранулярную языковую типологию
- На базе заявленной типологии разработать автоматический метод оценки токенов и применить его к словарям языковых моделей

Достигнутые результаты

В рамках данной работы был предложен новый метод дискурсивной сегментации текстов, введена и провалидирована новая гранулярная типология масштабирования языковых единиц, автоматизировано распознавание типов.

В первой части этого исследования был предложен метод дискурсивной сегментации текстов, который не требует дообучения, но сопоставим по качеству с лучшими подходами в этой области. Также, что более важно в контексте нашей общей исследовательской задачи, позволяет выделять крупные единицы гранулярности текста.

Далее, проведено объемное исследование выполнимости закона Ципфа при разных токенизациях и размерах словаря, выявлены условия декомпозиции распределения на два, характеризуемых противоположными семантическими и статистическими свойствами. Также в рамках дополнительного лингвистического исследования на выборке токенов, проведенном при участии экспертов лингвистов, выявлены количественные различия между выделенными группами токенов. На базе этих экспериментов выработана новаторская типология языковых единиц на пересечении статистических и семантических признаков, извлеченных при исследовании закона Ципфа.

В рамках полученной типологии спроектирована и реализована методология автоматической оценки лингвистической единицы на принадлежности к тому или иному классу, использующая архитектуру большой языковой модели. С помощью этого фреймворка получена классификация токенов в предобученном словаре языковой модели.

Таким образом, выработан новаторский глобальный подход к оценке единиц дискретизации языка, сочетающий статистические и лингвистические методики с обучением глубоких языковых моделей. Также, в продолжении исследования, предполагается использовать полученную типологию для улучшения методов токенизации. По промежуточным результатам данного исследования опубликована статья на NLP Воркшопе AAAI и готовится вторая публикация на EMNLP.

Структура работы

- В главе 1 рассмотрены релевантные работы в области дискурсивного парсинга, применимости закона Ципфа к обработке естественного языка, классические методологии токенизации и теоретико информационные оценки единиц и уровней языка
- В главе 2 предлагается методология извлечения крупных единиц гранулярности и описываются полученные с помощью нее результаты
- В главе 3 описываются эксперименты, проведенные в рамках закона Ципфа, анализируются полученные результаты, описывается разработанная типология, а также описывается методология автоматической оценки единиц и проведенные с ней эксперименты
- В последней главе делаются выводы по результатам всей работы, а также предлагаются направления для дальнейших исследований

1. Обзор Литературы

1.1. Методологии дискурсивного парсинга

Существует задача установления дискурсивных отношений в тексте: определение характера взаимоотношений между элементарными единицами дискурса EDU (elementary discourse units). Классическим представлением этих отношений является RST (rhetorical structure theory) [31] иерархическое дерево: в листьях располагаются EDU, а узлы более верхнего уровня содержат отношения между прилегающими EDU. Самый известный датасет такого вида это RST-DT [6]. Он содержит вырезки из Wall Street Journal. С точки зрения дискретизации текста на большие куски наибольший интерес представляет отношение смены темы.

Другое популярное представление это PDTB [37]. Обратимся к рисунку 1.1 [29].

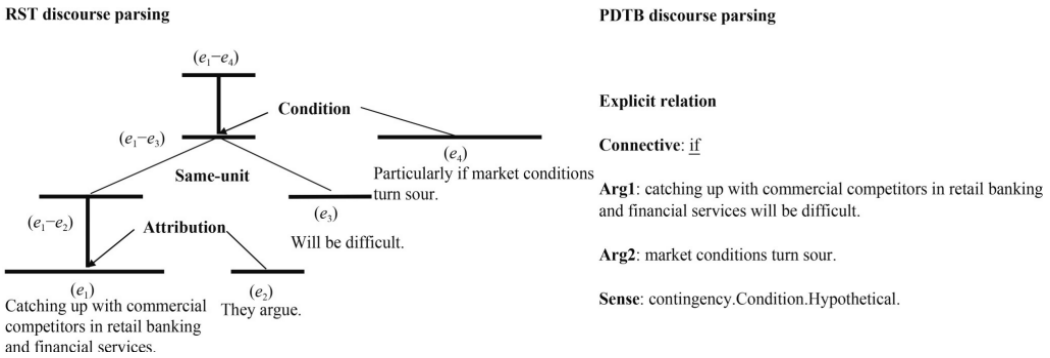


Рисунок 1.1: Сравнительная характеристика представления RST (слева) и PDTB (справа)

Основное различие заключается в том, что PDTB сконцентрировано вокруг локальной взаимосвязи между двумя поданными входами. В то время как RST дерево отражает в себе полную дискурсивную структуру текста на нескольких уровнях.

Условно задачу дискурсивного парсинга можно разделить на два этапа: дискурсивная сегментация и построение дерева. Дискурсивные связи можно разделить на две группы: явные и неявные. Явные выражены с помощью союзов, относительных местоимений, каких-то служебных

слов и как правило крайне хорошо определяются моделями. Другая же группа, неявных связей, представляет из себя комплексную структуру, вмещающую в себя гораздо больше узлов нижнего уровня и располагающуюся выше в дереве. Чаще всего EDU - это некоторые составляющие предложения с рядом условий: объекты, субъекты и дополнения к глаголу не являются отдельными EDU, а атрибуты (причастия и деепричастия) глагола являются, как и относительные предложения или предложения с сильными маркерами типа (because, so that и др.), Также к EDU относятся те элементы, которые разделяют структуру другой EDU на отдельные смысловые единицы.

Соответственно часть подходов оперирует с готовой разметкой дискурсивных единиц и концентрируется на построении самого дерева отношений. Саму задачу построения дерева можно описать так: нужно определить можно ли соединить два листа или два узла верхнего уровня, а также их атомарность и установить отношения. Несколько лет назад лучшие подходы в этой области строились на классических подходах к классификации. Например, наивный байесовский классификатор, который по явным связям, из которых свободно удалялись дискурсивные элементы, тренировался предсказывать неявные. Однако главное слабое место этого подхода - это сдвиг смыслов: явные и неявные отношения могут быть лингвистически совсем не похожи; и такое искусственное превращение явных отношений в неявные не корректно [39]. Также использовали легковесные нейронные сети, например, нейронная сеть прямого распространения (feedforward), использующая специфические для дискурса представления слов, которые были выучены моделью из большого количества явных пар аргументов [40]. На идейном уровне эти подходы пытались просто заучить отношения по размеченным данным, но так как неявных отношений в тексте очень много, то это не имело сильного успеха.

Тем не менее не сразу с появлением архитектуры трансформер [45] модели, основанные на ней, также стали применяться в дискурсивном парсинге. Это связано как с тем, что механизм внимания плохо справляется с длинными последовательностями, так и с проблемой самого представления RST и самого большого и применимого датасета на его основе RST-DT, которые доминировали по сравнению с PDTB в области довольно долго. Конкретно, эта сложность заключается в том, что RST-DT небольшой корпус, и большой нейронной сети не хватает данных, чтобы выучить представления слов. Тем не менее прорывом в области использования трансформерных архитектур для задачи дискурсивного парсинга явилась работа Kishimoto et al (2020) [21]. Авторы предлагают три метода:

- Предобученный под выявление дискурсивных отношений BERT [15]
- Предсказание явных связей на этапе обучения, чтобы потом использовать их в предсказании неявных
- На этапе настройки модели попытка без дообучения предсказать неявные связи (оказалась неудачной)

Идея такого подхода возникла из предшествующего исследования о попытке применить задачу NSP (Next Sentence Prediction): нужно по двум предложениям понять, правда ли, что второе предложение является продолжением первого, к выявлению неявных дискурсивных отношений [43]. В основе лежит идея о том, что, если модель в состоянии правильно предсказать является ли правый контекст продолжением левого, значит, она также в состоянии выявить связь между ними. Это влечет то, что она позволяет выявлять эти самые неявные дискурсивные отношения. В работе [21] в неразмеченном корпусе обнаруживаются (по ключевым словам, например) места, в которых возникают явные дискурсивные связи. Затем определяются промежутки, в которых находятся эти пары аргументов. Дальше, научившись на предобучении извлекать явные связи,

пытаются на дообучении научиться извлекать неявные. Однако авторы приходят к выводу, что для улучшения классификации неявных дискурсивных отношений необходимо учитывать внешние знания и лингвистические ограничения, поскольку аннотаторы-люди также полагаются на это общее знание [13]. Также существует исследование об иерархическом построении деревьев [22]. Деревья строятся следующим образом: 1) Сначала формируются деревья отдельно для уровня предложений, где листья это EDU; 2) Дальше строятся деревья для уровня параграфа, где листья — предложения; 3) Затем получают деревья и для уровня текста, где листья — это параграфы; 4) Потом меняются листья дерева уровня документа на деревья уровня параграфов, а листья уровня параграфов заменяются на деревья уровня предложений. У этого подхода главная проблема в выявлении явных дискурсивных сегментов на уровне текста, потому что явных связок между параграфами мало.

Тем не менее с точки зрения дискретизации текстов на крупные единицы нас интересует группа исследований, которая рассматривает не только задачу построения дерева по готовой разметке, но и само определение дискурсивных сегментов. Jiang et al. (2021) [20] выделяют единицы дискурса на уровне параграфов и строят над ними иерархическую структуру (формат RST), опираясь на идею из [22]. Как было сказано в начале раздела, задача разделяется на две: найти границы сегментов и построить дерево. Однако в этой работе особое внимание уделяется методологии сегментации. Для выделения сегментов представлена архитектура TM-BERT, которая основана на анализе связок из 3 последовательных параграфов: анализируются последовательно идущие параграфы и связь через один (1 +2, 2+3, 1+3), на каждую связку сфокусирован свой BERT. Архитектура представлена на рисунке 1.2

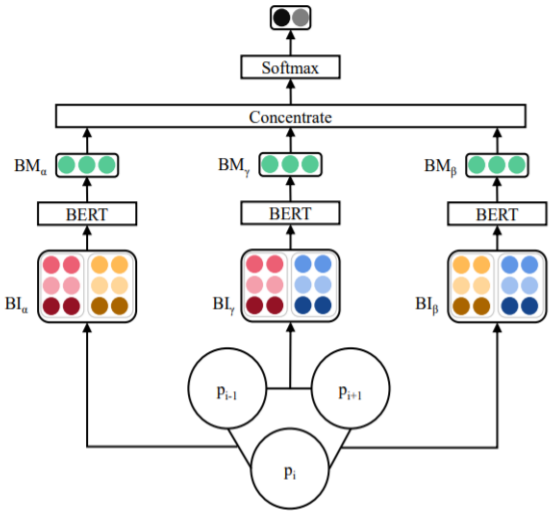


Рисунок 1.2: Архитектура TM-BERT: каждый из 3 «голов» фокусируется на свою связку из абзацев

Этот подход представляется излишне тяжеловесным и требующим много ресурсов при дообучении архитектуры из трех BERT. Также авторы не предоставляют отдельно метрики по сегментации и дискурсивному парсингу и лишь приводят изменения на конечной задаче, что неудобно с точки зрения воспроизводимости и сравнения по задаче сегментации. Исследователи отмечают, что использование их архитектуры для сегментации дает рост F-меры на финальной задаче построения дискурсивного дерева на 9.12. Так, важным представляется факт достаточно успешного использования BERT для задачи сегментации.

Далее стоит рассмотреть два взаимосвязанных исследования Koshorek et al. (2018) [23] и Xing et al. (2020) [48], которые концентрируются на методологии сегментации крупных текстовых единиц. Также в этих работах представлены конечные метрики и датасеты для сравнения именно на задаче сегментации. Это позволяет достичь лучшей воспроизводимости, не выпуская из рассмотрения задачу дискурсивного парсинга. Авторы более ранней работы [23] задали тренд на использование обучения с учителем и описали применение нейронной сети к задаче сегментации. Не менее важным вкладом также является

описание нового в области датасета: WIKI-727K включает в себя более 727 000 автоматически сегментированных документов из английской Википедии. Поскольку этот набор данных является большим, естественно образованным и охватывает различные темы, он предоставляет высокую степень обобщенности. Архитектура, представленная в данной работе это иерархическая нейронная модель на базе двунаправленной LSTM [19]. Подсеть нижнего уровня представляет собой двухуровневый двунаправленный LSTM, которая генерирует представления предложений: для каждого предложения s_i сеть берет на вход слова $w_1^i \dots w_k^i$ одно за другим и окончательное представление вычисляется путем макспулинга. На верхнем уровне находится сегментатор, который принимает на вход последовательности эмбедингов и пропускает через двунаправленную LSTM. На рисунке 1.3 представлена схема архитектуры.

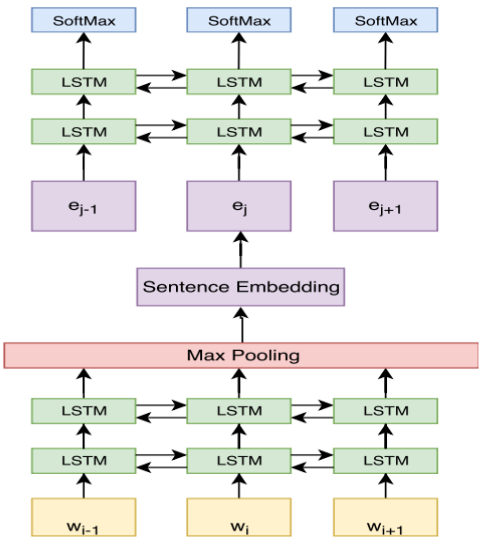


Рисунок 1.3: Схема архитектуры: подсеть для эмбедингов, за ней следует подсеть для предсказания сегментации.

Тестовые данные объемные, что не позволило авторам нормально сравниться с менее производительными архитектурами. Исходя из этого, они взяли набор из 50 случайно выбранных тестовых документов из основного датасета и провели сравнение на нем с теми архитектурами, с которыми нельзя напрямую сравниться на WIKI-727K. Более того, на этих

50 документах также предоставляется человеческая разметка. При сравнении использованы также некоторые искусственные датасеты, но как видно будет из дальнейшего анализа основой для сравнения последующих архитектур является именно WIKI-727K. В качестве метрики используется P_k [1]. Выражается она следующей формулой:

$$P_k(ref, hyp) = \sum_{i=0}^{n-k} \delta_{ref}(i, i+k) \neq \delta_{hyp}(i, i+k)$$

Где δ это индикаторная функция границы сегмента. Метрика измеряет вероятность несовпадений между истинными сегментами (ref) и прогнозами модели (hyp) в пределах скользящего окна k. Поскольку P_k является штрафующей метрикой, меньший показатель указывает на лучшее качество.

По сравнению с предыдущими лучшими подходами, авторам удалось достичь значительного роста качества, что видно из таблицы 1.1.

Таблица 1.1: Значения метрики P_k на пяти датасетах, основной из которых WIKI-727K для ряда старых подходов и авторского подхода. Жирным выделен лучший результат.

P_k variant	WIKI-727K	WIKI-50	CHOI	CITIES		ELEMENTS	
	sentences	*sentences*	*sentences*	*sentences*	*words*	*sentences*	*words*
(Chen et al., 2009)	-	-	-	-	22.1	-	**20.1**
GraphSeg	-	63.56	**5.6-7.2**	39.95	-	49.12	-
Our model	22.13	**18.24**	26.26[3]	**19.68**	**18.14**	**41.63**	33.82
Random baseline	53.09	52.65	49.43	47.14	44.14	50.08	42.80
Human performance	-	14.97	-	-	-	-	-

Развивая идею Koshorek et al. (2018) [23], позднее была предложена более сложная архитектура [48], сочетающая механизм внимания и LSTM. Во-первых, авторы добавляют вспомогательную задачу, связанную с когерентностью, чтобы модель училась выделять более информативные скрытые состояния для всех предложений в документе. Более конкретно, цель модели — определять меньшую связность для предложений из

разных сегментов и большую связность для предложений из одного сегмента. Во-вторых, они улучшают моделирование контекста, используя ограниченный механизм внимания [46], что позволяет их модели обращать внимание на локальный контекст и лучше использовать информацию от ближайших соседей каждого предложения (по отношению к окну явно фиксированного размера k). Архитектура представлена на рисунке 1.4. Авторам удалось почти в два раза улучшить результат Кошорека.

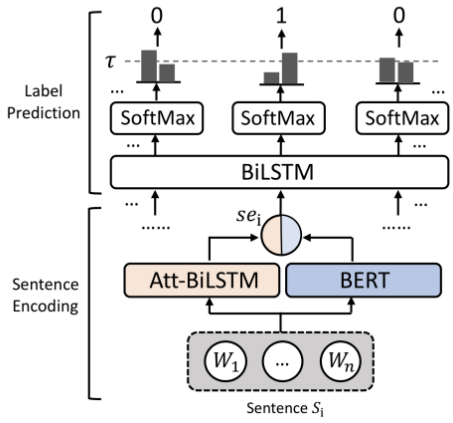
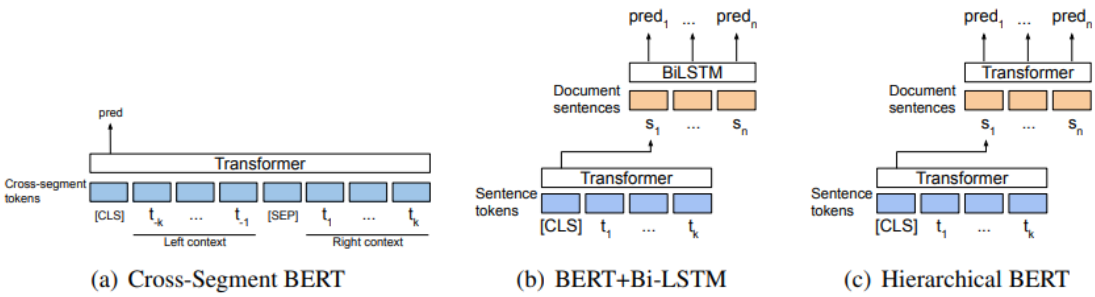


Рисунок 1.4: Архитектура модели, сочетающая BERT и Attention + BiLSTM; se_i это полученный для предложения S_i эмбеддинг.

Другой подход к вопросу о необходимости проектировать такие сложные архитектуры, как описано выше, реализован в работе Lukasik et al. (2020) [30]. На рисунке 1.5 представлены схемы трех представленных в статье архитектур.



(a) Cross-Segment BERT (b) BERT+Bi-LSTM (c) Hierarchical BERT

Рисунок 1.5: Кросс-сегментный BERT (слева) оперирует с локальным контекстом, окружающем потенциальный разрыв сегмента: k лексем слева и k лексем справа. В модели BERT+Bi-LSTM (в центре) BERT

используется для получения представлений слов, а уже двунаправленный LSTM отвечает за сегментацию, иерархический BERT (справа) похож на предыдущий подход только вместо LSTM используется трансформер.

Особый интерес представляет первая модель, так как она отличается простотой и в то же время успешно оперирует только с локальным контекстом. Предположительно механизм внимания вкупе с предобучением BERT на задаче NSP дает достаточно обобщающей способности, чтобы корректно определять семантические сдвиги внутри крупных единиц гранулярности. Наряду с P_k авторы также считают полноту, точность и F-меру. Из таблицы 1.2 видно, что cross-segment BERT сопоставим по качеству с более сложной и требующей больше ресурсов для обучения третьей моделью.

Таблица 1.2: Значения метрик полноты, точности и F-меры для трех датасетов. В первом блоке расположены предыдущие подходы. Во втором блоке 3 авторские архитектуры.

	Wiki-727K			RST-DT			Choi	
	Precision	Recall	F1	Precision	Recall	F1	F1	P_k
Bi-LSTM (Koshorek et al., 2018)	69.3±0.1	49.5±0.2	57.7±0.1	-	-	-	-	-
SEGBOT (Li et al., 2018)	-	-	-	91.6	92.8	92.2	-	0.33
Bi-LSTM+CRF (Wang et al., 2018)	-	-	-	92.8	95.7	94.3	-	-
Cross-segment BERT 128-128	69.1±0.1	63.2±0.2	66.0±0.1	92.1±0.8	**98.0±0.4**	95.0±0.5	**99.9±0.1**	**0.07±0.04**
BERT+Bi-LSTM	67.3±0.1	53.9±0.1	59.9±0.1	**94.4±0.5**	96.0±0.4	**95.2±0.3**	99.8±0.1	0.17±0.06
Hier. BERT	**69.8±0.1**	**63.5±0.1**	**66.5±0.1**	93.8±0.7	96.7±0.5	**95.2±0.4**	99.5±0.1	0.38±0.09
Human (Wang et al., 2018)	-	-	-	98.3	98.2	98.5	-	-

Эта статья является значимым этапом в использовании исходной обобщающей способности больших языковых моделей при решении задачи дискурсивной сегментации. Это, в свою очередь, вносит вклад как в исходную более общую задачу дискурсивного парсинга, так и в границы применимости с разным уровнем дообучения языковых моделей при решении узкоспециализированных задач.

1.2. Классические методы токенизации

Двигаясь от дискретизации текста на крупные смысловые единицы, мы переходим к рассмотрению классических методов токенизации. Стоит отметить, что речь пойдет об архитектурах, основанных на идее

токенизации на подслова, так как в свое время именно эти способы решили проблему с отсутствием в обучающей выборке тех или иных редких многосоставных слов и заняли основное положение в области. Основополагающей идеей, предложенной исходно в трудах по кодированию и производной от алгоритма Хаффмана, является BPE [16, 42]. Суть метода заключается в том, чтобы последовательно объединять наиболее часто встречающиеся пары символов (байтов) в корпусе текстовых данных до тех пор, пока не будет достигнуто требуемое количество токенов. Например, если в корпусе часто встречается последовательность символов "th", то она может быть объединена в новый токен "<th>". Затем такой же подход может быть применен к другим часто встречающимся парам символов, таким как "an", "in", "er" и т.д. В результате получится словарь токенов, состоящий из новых токенов и оригинальных символов. Отсюда видно, что BPE успешно справляется с обработкой неизвестных слов. Тем не менее при обработке больших текстовых корпусов, он может генерировать слишком большой словарь, поэтому, как правило, при работе с языковыми моделями размер словаря искусственно ограничивается до приемлемого с точки зрения скорости оперирования данными значения. Это, впрочем, также может являться узким местом языковой модели. Следующий алгоритм это Wordpiece [47]. Его реализация похожа на BPE, однако в отличие от BPE его процедура слияния основана на максимизации вероятности обучающих данных после их добавления в словарь. Этот алгоритм токенизации применяется в широко используемых моделях, таких как BERT, DistilBERT, Electra[15, 41, 8]. Другим базовым алгоритмом является Unigram [24], который инициализирует свой базовый словарь значительным количеством символов и постепенно сокращает каждый символ для получения меньшего словарного запаса. Базовый словарь может, например, соответствовать всем предварительно помеченным словам и наиболее распространенным

подстрокам. Процедура сокращения основана на моделировании распределения над сегментами. Unigram не встраивается непосредственно в языковые модели, а используется совместно с SentencePiece [25]. Алгоритм SentencePiece, с другой стороны, рассматривает данные как необработанные входные данные, включая пробел в используемый набор символов. Затем внутри SentencePiece применяется алгоритм BPE или Unigram для построения соответствующего словаря. Примерами моделей, использующих SentencePiece, являются ALBERT [26] и XLNet [50]. Также недавно появился метод, соединяющий идеи BPE и Unigram: BPE-dropout [38]. В классическом BPE бывают проблемы с интерпретацией составных частей сложных слов, а также с переобучением под конкретные токены. Авторы предлагают добавлять случайный выбор (дроп) при замене наиболее частых последовательностей символов на новый токен. На рисунке 1.6 представлена иллюстрация к работе алгоритмов.

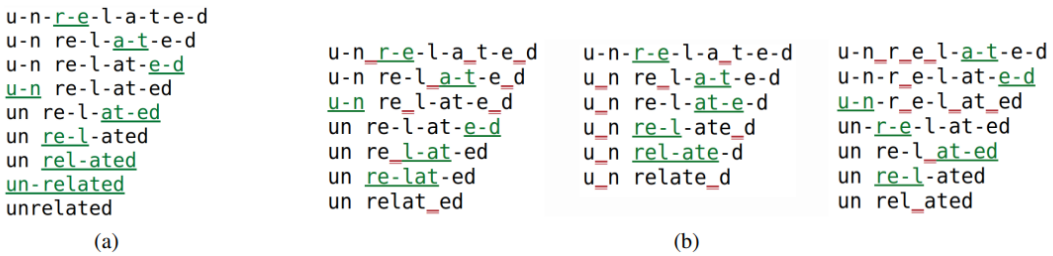


Рисунок 1.6: Слияние в алгоритме BPE (слева) и в BPE-dropout (справа).

Авторы отмечают, что данная методология превосходит BPE и Unigram по качеству на BLEU [34] за счет уменьшения переобучения и повышения обобщающей способности. Однако ясным представляется, что перечисленные алгоритмы явно полагаются исключительно на статистические признаки в реализации механизма обучения токенизатора. К сожалению, ни в одном из приведенных исследований не рассматриваются семантические свойства уже полученного словаря и не делается попытка использовать семантически ориентированный подход в механизме обучения.

1.3. Закон Ципфа в естественном языке

Закон Ципфа — это эмпирический закон, который описывает статистическое распределение слов в тексте естественного языка. Согласно закону Ципфа [52], частота встречаемости слова в тексте обратно пропорциональна его порядковому номеру в ранжированном списке слов текста. Другими словами, наиболее часто встречающееся слово в тексте (обычно это артикль, союз или предлог) встречается примерно в два раза чаще, чем второе наиболее часто встречающееся слово, в три раза чаще, чем третье слово, и так далее. Закон Ципфа применим к любому тексту на естественном языке, независимо от его длины или темы. Он описывает общую закономерность в распределении слов в тексте, но не объясняет причины этой закономерности.

Многие исследователи изучали значение закона Ципфа для различных проблем NLP. Например, [49] предполагает, что закон Ципфа способствует раннему освоению языка, и использует закон Ципфа для отличия использования языка человеком от биоакустики других видов, например шимпанзе. Также существует исследование на пересечении естественного языка и программного кода, которое обнаружило, что распределение лексем в исходном коде Java следует закону Ципфа [51]. Эти статьи позволили нам ознакомиться с конкретными границами применимости некоторых проявлений закона Ципфа в NLP.

Тем не менее, есть группа исследований, которые оказали значительное влияние на эту работу. Существует два ранних исследования, которые независимо и с разных сторон изучали точку декомпозиции в распределении Ципфа для достаточно большого корпуса естественного языка. Это характеризуется разной выполнимостью закона Ципфа при разной наполняемости словаря. Во-первых, для отдельных слов в корпусе закон Ципфа выполняется не полностью. Точнее, он справедлив только для высокочастотных слов, а затем распределение начинает

смещаться от обратного с коэффициентом -1 в сторону обратно квадратичного. На рис. 1.7 прослеживается отклонение закона при увеличении номера позиции в ранжированном списке.

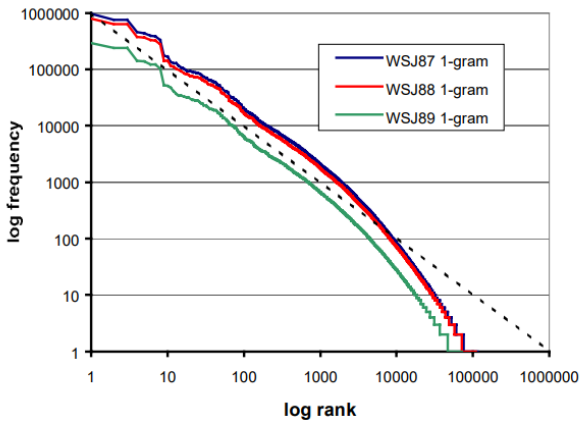


Рисунок 1.7: Кривые закона Ципфа для unigram для корпуса, основанного на вырезках из Wall Street Journal.

Однако если добавить n-граммы (то есть сочетания слов при n=2, 3.. etc), то выполнимость закона Ципфа становится полной. Эти наблюдения представлены в работе Le et al. (2002) [27]. С другой стороны, авторы Cancho et al. (2001) другого исследования [4] рассматривают декомпозицию закона Ципфа на два, с разными параметрами. Исследователи указывают что первая часть распределения скорее следует чистому закону Ципфа с коэффициентом -1, а вот более «тяжелая» вторая часть начинает сдвигаться ближе к обратно квадратичному. Вот эта вторая часть распределения медленнее затухает и чем больше корпус, из которого извлекаются слова, тем больше совокупное распределение сдвигается именно в сторону второго «тяжелого». Хотя они по-прежнему сосредоточены на статистических особенностях распределения Ципфа, в разделе обсуждения они предполагают, что первое распределение охватывает слова базовой коммуникации, характеризующиеся многофункциональностью, а второе, в свою очередь, относится к словам узкоспециализированной коммуникации. В развитие этих идей в [11] обнаружили, что многие натуральные системы не демонстрируют

истинное поведение по степенному закону, поскольку они неполны или не соответствуют условиям, при которых можно было бы ожидать появления степенного закона. Это исследование особенно актуально в плане введения понятия когерентного подмножества, применимого к таким единицам, которые распределены относительно чистого степенного закона. Мы считаем идеи, освещенные в этих работах, ключевыми отправными точками для анализа методов токенизации с использованием особенностей распределения закона Ципфа.

1.4. Теоретико информационные оценки лингвистических единиц

Существуют и другие методологии оценивания лингвистических единиц; в этом обзоре мы сконцентрируемся на тех из них, что основаны на теории информации, потому что эта систематика является одной из ключевых в машинном обучении и в обработке естественного языка, в частности. Наиболее важными мы считаем те исследования, которые соотносят между собой теоретико информационные оценки и описанный в предыдущем разделе закон Ципфа, так как формирование взаимосвязи между статистическими, лингвистическими и теоретико информационными свойствами языковых объектов представляется нам важным направлением в области, напрямую влияющем на создание новых направлений в рамках улучшения общей производительности языковых моделей через более фундаментальное понимание единиц, которыми они оперируют. Для начала, в [44] авторы утверждают, что признаком эффективных представлений является то, что частотные распределения следуют степенным законам. В [5] показывается, что одного предположения о совместной вероятности слова и значения достаточно для вывода закона Ципфа о частоте значений, и утверждают, что это предположение может быть оправдано как результат смещенного

случайного блуждания. Другая группа исследователей Pimentel et al. (2021) и Nikkarinen et al. (2021) [33, 35, 36] выпустила целую серию работ, посвященную анализу оптимизированности языка.

В наиболее ключевой из которых [35] демонстрируют, что естественные кодирования языка ближе к неоптимизированным (в ципфианском смысле), чем к максимально сжатым. Есть закон сокращения, выводимый из закона Ципфа (law of abbreviation), он гласит: чем длиннее слово, тем реже используется. Однако известны примеры коротких редких слов и длинных частотных. Соответственно можно утверждать, что наш естественный язык недостаточно оптимален с точки зрения эффективности в смысле соответствия ципфовскому закону сокращения. У естественного языка есть разнообразные ограничения. Например, морфологические: многие слова формируются из подслов и не всякая конкатенация таких подслов дает осмысленное слово. А также графотатические: на письме для большинства языков невозможно встретить слово, состоящее из пяти согласных и т.п. Авторы исследуют теоретико информационную стоимость различного рода ограничений в синтаксически и семантически многоплановом наборе языков, выбрав в качестве меры оптимальности указанное соответствие закону сокращения. Они приходят не только к выводу о ципфовской неэффективности натурального представления, но и к тому, что именно морфологические и графотатичекие признаки являются ключевыми для реконструкции языка и дают основную неоптимальность.

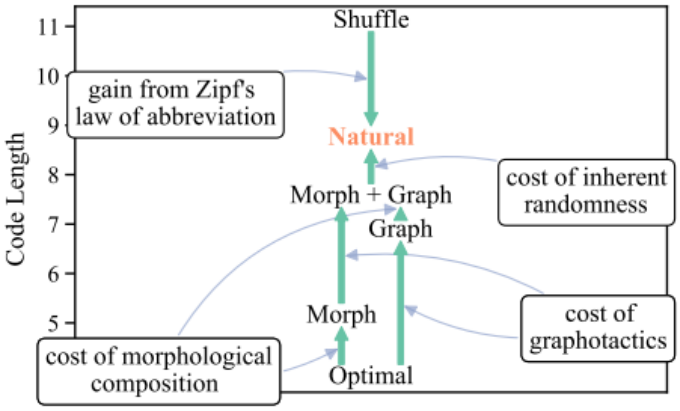


Рисунок 1.8: Средняя длина кодированного языка при использовании разных схем ограничений на примере финского. Расстояние между базовыми значениями можно рассматривать как стоимость каждого добавленного ограничения к системе.

Более того, в сопутствующей работе [33] та же исследовательская команда заявляет важность правильного моделирования распределения unigram; а любой подход, который приписывает нулевую вероятность любой словоформе вне словаря, дает отрицательно смещенные вероятности, поэтому при моделировании данного распределения необходимо балансировать между словоформами с частотами (токены) так и уникальными словоформами (типы). Это нужно для того, чтобы можно было эффективно промоделировать и редкие словоформы и атипичные невстречающиеся в языке, которые могли быть сгенерированы моделью и занесены в общий словарь при обучении. Кроме того, есть еще одна работа [36], посвященная оценке меры внутриязыковой произвольности. В частности, авторы, используя теорию информации, моделируют универсальные распределения форм слов и анализируют взаимную информацию между значением слова и формой. Они также определяют, какие понятия демонстрируют более сильную связь между произвольной формой и значением и какие типы форм встречаются в них чаще. Эти работы дают поле для дальнейших исследований в выявлении связи между семантической и энтропийной оценкой лексем в словаре языковой модели. Наконец, особняком стоит работа [32] о связи между выполнимостью

гипотезы о равномерной плотности информации (UID) и метриками в задачах естественного языка. UID предполагает, что пользователи языка предпочитают высказывания, структурированные таким образом, чтобы информация распределялась равномерно по сигналу. Авторы показывают, что по метрике лингвистической приемлемости (acceptability) наиболее высокие результаты показывают те примеры, что лучше соответствуют UID и что UID хорошо предсказывает процессинговые усилия высказывания.

1.5. Выводы

- Существует большое количество подходов к задаче дискурсивного парсинга. При этом лишь незначительная группа работ сконцентрирована именно на подзадаче сегментации
- Большая часть из предлагаемых в дискурсивном парсинге и дискурсивной сегментации подходов — это сложные архитектуры, требующие значительного дообучения
- Не существует исследований, включающих характерные крупные элементы текстовых последовательностей в систематическую типологию единиц дискретизации языка с точки зрения изучения масштабов гранулярности, которыми оперируют известные большие языковые модели
- Существует значительное множество подходов к дискретизации текстов на мелкие единицы так называемые токены. На базе этих подходов реализованы алгоритмы, использующиеся при формировании словаря языковой модели. Тем не менее в них не исследуется семантическая составляющая получаемых единиц
- Рассмотрены подходы к оцениванию языковых элементов на пересечении закона Ципфа, теории информации и обработки естественного языка.

- Выявлены направления для выработки собственной гранулярной типологии, в частности область изучения когерентных подмножеств текстовых единиц

2. Дискурсивная сегментация

В этой главе предложен метод дискурсивной сегментации текстов, не требующий дообучения и показывающий сопоставимые результаты с лучшими подходами в области, описаны проведенные эксперименты, а также выявленные в ходе работы недостатки классических для области метрик. Также воспроизведены архитектуры предложенные [30] и проведены эксперименты с ними.

2.1. Описание методологии

Как было рассмотрено в разделе 1.1 обзора литературы, текущие подходы к дискурсивной сегментации даже если используют языковые модели, то делают это в архитектурно переусложненных конфигурациях, и все равно вынуждены производить значительное дообучение под задачу. Чтобы решить эту проблему, мы решили обратиться к высокой обобщающей способности больших языковых моделей, например, BERT [15] и предложить методологию, которая вычислительно будет намного менее затратной.

Итак, BERT предобучается таким образом, чтобы учитывать двусторонний контекст; данная архитектура учится на двух задачах: MLM (Masked Language Modelling): восстановление скрытого контекста и NSP (Next Sentence Prediction). Задача NSP была сформулирована как задача бинарной классификации: модель обучается для того, чтобы отличать реальное следующее предложение от случайно выбранного предложения из корпуса. Таким образом, предобученная модель уже содержит в себе определенное ожидание относительно того, как выглядит следующее

предложение, если ей дан префиксный контекст. Предложения, находящиеся внутри одного дискурсивного сегмента, фактически являются продолжениями контекста друг друга. Там, где происходит смена сегмента, нарушается связь между префиксным и суффиксным контекстом. Таким образом, по низкой уверенности модели в том, что правый контекст является продолжением левого, можно определять границу смены сегмента. Способность модели успешно (или неуспешно) определять эти границы, опираясь лишь на предобученность на задаче NSP, дает возможность выделить характерные верхнеуровневые элементы текста в отдельную типологическую единицу. В будущем представляется интересным вопрос рассмотрения общей успешности процессинга модели на этом уровне в сравнении с более низкоуровневыми элементами на разнообразных конечных задачах.

На базе заявленной методологии была проведена серия экспериментов для проверки валидности, высказанных выше предположений. Во-первых, эксперименты проведены на предложенном [23] датасете, собранном из Википедии. Так как авторы уделили большое внимание препроцессингу данных, очистке их от зашумляющей и снижающей качество работы модели ненужной информации (внетекстовые элементы, спецзначки и т.п.) и предоставляют уже очищенный и подготовленный датасет с разметкой границ сегментации, фактически можно пользоваться им без дополнительной обработки. Все последующие работы, которые замеряли качество модели на этом датасете, также особо подчеркивают необходимость использовать предоставленные данные как есть для полной воспроизводимости. Модель движется скользящим окном (при этом параметр размера окна, а также соотношения длины между левым и правым контекстом варьируется в различных конфигурациях экспериментов) по эмбеддингам предложений, и для получения вероятности предсказания к полученным логитам применяется софтмакс на выходе. Соответственно,

мы получаем две вероятности: 1) того, что правый контекст продолжает левый 2) того, что правый контекст НЕ продолжает левый. При высоком показателе второй вероятности отсутствие связи между предложениями сигнализирует о том, что дискурсивная единица закончилась и началась новая, то есть левый контекст является закрывающим для одного дискурсивного сегмента, а правый - открывающим для другого. Были проведены эксперименты с разными пороговыми значениями вероятности модели, при которой производилось отнесение к тому или иному классу (то есть наличию или отсутствию смены дискурсивного сегмента). В ходе этих экспериментов был обнаружен феномен “самоуверенности модели”: она проявляет высокую уверенность, близкую к 1, по одному классу и низкую, близкую к 0, по второму; таким образом, ситуаций, когда она дает примерно равные или даже значительно отличающиеся, но не такие предельные показатели, просто не возникало, поэтому вариация пороговых значений не влияет на результаты. Дополнительная группа экспериментов, посвященных увеличению только префиксного контекста, не показала приемлемых результатов и подтвердила предположение о том, что для модели является критически важным видеть обе стороны вокруг предполагаемого разрыва сегмента. Это может быть связано в том числе с тем, что распределения слов до и после истинного разрыва сегмента совершенно разные. Тем не менее языковые модели оперируют не только с распределением последовательностей слов при определении сдвига семантики, которая возникает на границе сегмента дискурса. Этот вопрос заслуживает отдельного, более детального рассмотрения и тесно связан с более глобальным об эффективных единицах, которыми оперирует модель. Тем не менее основные эксперименты со скользящим окном показали следующие результаты:

1. Увеличение окна, позволяя захватывать больше информации в модель, повышает результаты. Однако важно подчеркнуть, что

слишком большое окно, значительно превышающее половину средней по корпусу длины сегмента, не дает выигрыша в метриках.

2. В классической в этой области метрике P_k наблюдаются аномалии (более подробно обсуждается в разделе 2.2) и имеет смысл наряду с ней также смотреть на классические точность, полноту и F-меру. У P_k чем ниже значение, тем лучше качество модели, в отличие от F-меры, это надо учитывать при сравнении результатов.
3. Обобщающей способности модели, выработанной в ходе предобучения под NSP вполне достаточно, чтобы показывать сопоставимые с дообученными и, что более важно, сложно устроенными архитектурами. Результаты представлены в таблице 2.1.

Таблица 2.1: Значения F-меры и метрики P_k для рассмотренных в разделе 1.1 литературного обзора и для нашей методологии. Жирным выделен лучший результат. Также приведены значения метрики P_k для людей, однако эти анотаторы производили разметку на небольшой выборке документов, и поэтому показатель не может использоваться для прямого сравнения, но представляет определенную степень иллюстративности.

Подход	F-мера	P_k
BERT + BiLSTM	59.9	**9.7**
Иерархия BiLSTM	57.7	18.24
PV	31.1	28.1
CNN	42.1	21.9
SEC	28.4	24.5
Cross-segment BERT	66.01	-
Hierar. BERT	**66.5**	-

Эта работа	57.06	22.7
Человек	-	14.97

2.2. Аномалии в метриках

Напомню, что P_k представляет из себя следующую формулу:

$$P_k(ref, hyp) = \sum_{i=0}^{n-k} \delta_{ref}(i, i+k) \neq \delta_{hyp}(i, i+k)$$

В качестве стандартной настройки, которая использовалась в предшествующих исследованиях, размер окна k — это половина средней длины сегмента ref.

Во всех предшествующих работах, где производится сравнение моделей дискурсивной сегментации, эта метрика использовалась без какого-либо переосмысления, просто как принятый стандарт. Тем не менее в ходе экспериментов мы обратили внимание на следующие ее особенности, которые могут сдвигать понимание качества модели:

1. P_k учитывает близость промаха модели, таким образом, если граница сегмента определена неправильно в небольшой окрестности реальной точки сдвига, то штраф будет сильно меньше, чем при более удаленной ошибке. С одной стороны, метрика сознательно конструировалась именно таким образом, и это действительно разумно, с другой - при наличии несбалансированных классов, сдвигов в данных и т.п., она становится сильно менее строгой, чем та же F-мера, и не в полной степени отражает качество модели.
2. P_k подвержена сдвигам в оценке: штраф за ложноотрицательные результаты более серьезный, чем за ложноположительные
3. Количество смен сегментов значительно меньше, чем количество согласованных предложений внутри одного сегмента. Это

происходит, по-видимому, в силу специфики структуры языка на верхнем уровне: смена крупных единиц происходит реже. В работе [30] также подчеркивается (без анализа) недостаточность метрики P_k и предлагается использовать еще и F-меру.

В связи с этим удалось сделать следующие выводы:

1. Картина более полная, если предоставлять обе эти метрики
2. Требуется провести дополнительный анализ соотнесенности метрик между собой и возможно выработать некоторый новый подход к оцениванию качества дискурсивной сегментации, сочетающий плюсы обоих метрик и устраняющий их недостатки. Как один из возможных вариантов: взвешивание вклада каждого класса при вычислении функции потерь модели. При этом сэмплирование примеров вниз (до количества по нижнему классу) для тех архитектур, которые дообучаются под задачу, может быть фатально для общего качества.

2.3. Воспроизведение результатов Lukasik

Работа [30] попала в фокус внимания позже, чем другие лучшие подходы в области. Изначально в нашей работе после полученных достаточно хороших результатов на обычном BERT с задачей NSP планировалось как второй этап все-таки дообучить его под дискурсивную сегментацию и оценить, насколько это повышает показатели метрик и как это соотносится с количеством времени и графических процессоров, требующихся для дообучения. Тем не менее исследование [30] закрывает этот вопрос, так как несмотря на то, что изначально авторы концентрировались на двух достаточно сложных архитектурах в основе, которых лежит BERT (см. рис. 1.5), они неожиданно обнаружили, используя просто дообученный cross-segment BERT как бейзлайн, что он совсем незначительно уступает в качестве более сложным из

предложенных ими архитектур. К сожалению, несмотря на явный запрос с моей стороны, авторы статьи не предоставили код, поэтому пришлось воспроизводить с нуля. Мы сконцентрировались на cross-segment BERT, так как он наиболее близок по архитектурной составляющей к тому, что планировали в развитии своих идей делать мы и отвечает требованию легковесности архитектуры. Тем не менее полное дообучение BERT на таких больших данных не представляется возможным в силу ограниченности вычислительных возможностей, имеющихся у нас. Авторы в свою очередь имели доступ к большому количеству TPU. В связи с этим мы обучали в течение совсем небольшого количества эпох и на куске датасета, поэтому полученные результаты нельзя считать полным воспроизведением. Мы провели эксперименты с несколькими конфигурациями количества токенов, подающихся на вход модели: при увеличении качество также растет, однако следует отметить, что вопрос увеличения количество подаваемого контекста, а также параметров модели и т.п конфигурационных элементов вкупе с анализом отношения затрачиваемых на обучение ресурсов к росту метрик пока не до конца исследован и, возможно, следует ожидать от той же исследовательской команды продолжения в этой области. В таблице 2.2 приведены полученные результаты.

Таблица 2.2: Результаты воспроизведения cross-segment BERT с разной длиной контекста. В качестве метрики F-мера.

Длина контекста	Эпоха 1	Эпоха 2	Эпоха 3
16	64.5	67.6	69.1
32	69.4	71.2	71.9
64	74.4	75.2	75.5
128	74.5	75.7	76.3

Как продолжение исследований в данной области интересным представляется подход, сочетающий нашу методологию и cross-segment BERT, а также техники дистилляции.

2.4. Выводы

- BERT предобученный под NSP без дообучения под задачу дискурсивной сегментации показывает сопоставимые с более сложными SOTA архитектурами результаты и значительно превышающие классические подходы
- Существуют определенные аномалии в классической метрике, нами предложен анализ причин возникновения этих аномалий и возможные решения
- Частично воспроизведены результаты наиболее идеологически близкой к нашей работе статьи, установлено, что дообучение даже самой простой BERT-based архитектуры требует значительных вычислительных ресурсов и, следовательно, по этому параметру наш способ вкупе с достаточно высокими показателями метрик конкурентоспособен SOTA
- Есть основания выделить, получаемые в ходе дискурсивного парсинга сегменты в отдельную единицу типологии элементов, которыми модель способна оперировать эффективно. Однако требуются дальнейшие более детальные исследования как операционной способности языковой модели, так и характеристик самой единицы и ее соотношения с другими более низкоуровневыми единицами

3. Закон Ципфа и семантически обусловленная процедура токенизации

В этой главе описываются эксперименты с токенизациями, проведенные в рамках закона Ципфа, анализируются их результаты, вводится новая типология текстовых единиц, представляется методология автоматической оценки на принадлежность единицы к тому или иному классу и анализируются границы ее применимости.

3.1. Причины выбора методологии

Развивая идею масштабирования текстовых единиц, мы обратились к исследованию более низкоуровневых единиц. В качестве базиса был выбран закон Ципфа и извлекаемые на его основе характеристики элементов; в разделах 1.3–1.4 литературного ревью показывается продуктивность данной методологии и ее широкая применимость к задачам обработки естественного языка. По мере увеличения размера словаря или объема данных в нем наблюдается явление, когда некоторые лингвистические элементы имеют очень низкую частоту встречаемости, что делает их малоинформативными для моделей, основанных на распределении слов. Большинство современных методов обработки естественного языка ограничивают словарь модели, исключая такие редкие элементы и оперируя токенами. Однако, поскольку модели обучаются на текстовых корпусах, которые подчиняются закону Ципфа, возникают интересные вопросы о том, как, в том числе по семантическому признаку, элементы могут быть сгруппированы в зависимости от их соответствия закону Ципфа:

1) Как конкретная процедура токенизации пересекается с законом Ципфа?
2) Существуют ли четкие различия между частыми "головными" и нечастыми "хвостовыми" лексемами?

3) Могут ли эти различия повлиять на производительность модели и быть критически важными для некоторых аспектов задач NLP, которые может решить модель?

В попытках ответить на эти вопросы мы исследуем, как эмпирический факт "тяжелого хвоста" распределения Ципфа мешает методам, основанным на идее распределительной семантики.

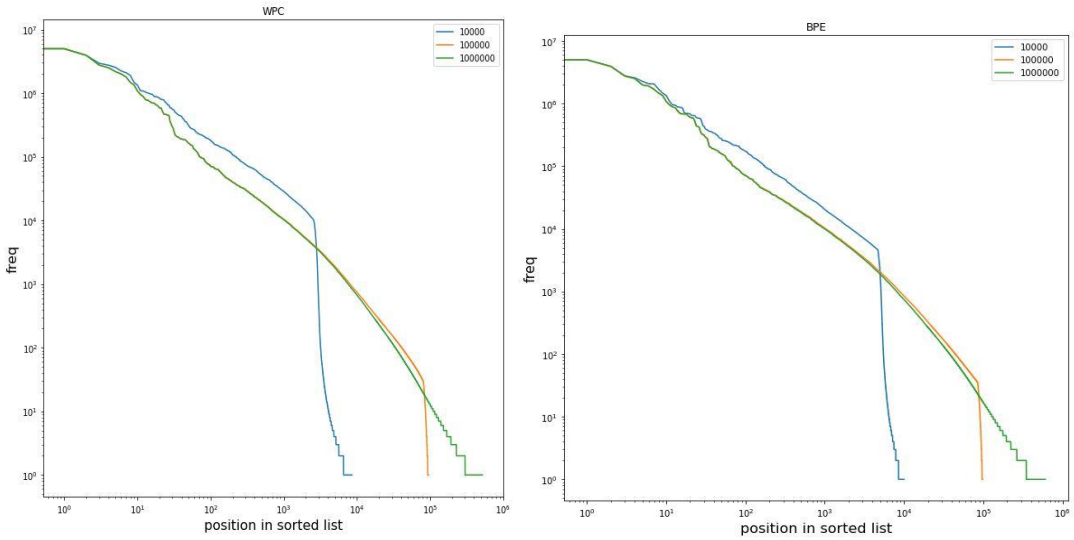
3.2. Эксперименты с токенизациями на основе закона Ципфа

В данном разделе мы рассказываем про методику и результаты изучения поведения распределения ранга-частотности для наборов токенов с разными максимальными длинами (определенными размером словаря и алгоритмом токенизации) и показываем, что это распределение хорошо аппроксимируется суперпозицией двух законов Ципфа для двух различных, внутренне связанных (иными словами, когерентных) наборов токенов.

Эксперименты проведены с тремя различными алгоритмами токенизации: Byte Pair Encoding (BPE) [16], WordPiece [47] и Unigram [24]. Более подробно эти алгоритмы обсуждаются в разделе 1.2 обзора литературы. Обучение токенизаторов производится на большом естественно составленном датасете wikitext-103. Он представляет собой корпус из более чем 100 миллионов токенов, извлеченных из набора проверенных хороших и выдающихся (категории) статей на Википедии. По сравнению с предобработанной версией Penn Treebank (PTB), набор данных WikiText-2 вдвое больше, а WikiText-103 более чем в 110 раз больше. В наборе данных WikiText также присутствует гораздо больший словарь и сохраняются исходный регистр, знаки препинания и числа, которые удаляются в PTB. Мы не обнаружили существенных различий в поведении частоты токенов по отношению к их рангу. До определенного предела

размера словаря распределение ранга-частотности следует закону Ципфа независимо от выбранного алгоритма токенизации.

На рисунке 3.1 представлены результаты выполнимости закона Ципфа для разных словарей и алгоритмов токенизации.Все они демонстрируют поведение, подобное закону Ципфа. Мы полагаем, что BPE является наиболее наглядным алгоритмом в силу своей природы. При увеличении размера словаря BPE создает более длинные токены, так как он добавляет наиболее частую пару существующих токенов в словарь в качестве нового токена. Это приводит к простому следствию: более значительный размер словаря ведет к большей максимальной длине токена. В дальнейшем, если алгоритм токенизации не указан явно, мы используем BPE.



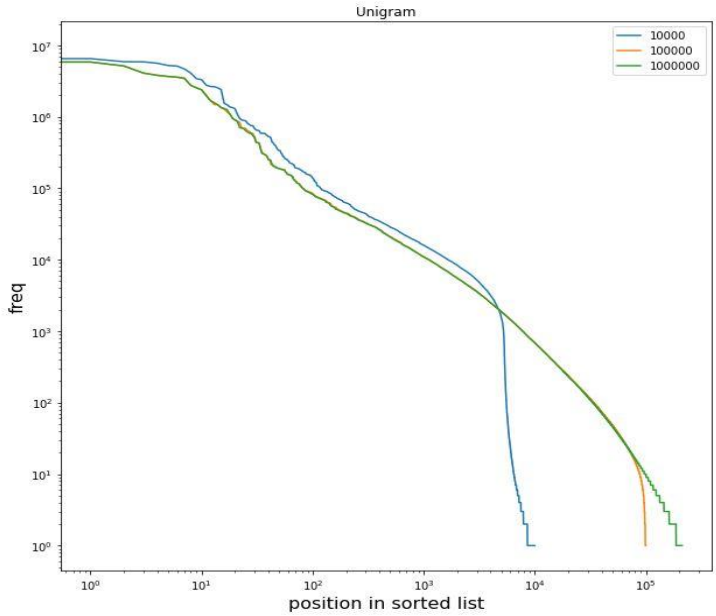


Рисунок 3.1 Распределение ранг частотности на Wikitext-103 для BPE, Wordpiece и Unigram для размеров словаря 10 000, 100 000, 1 000 000.

Мы провели серию экспериментов с различными размерами словарей, начиная от нескольких тысяч (что дает токенизацию на уровне символов, N-грамм и слов) до нескольких миллионов (что неизбежно добавляет в словарь токены, состоящие из нескольких слов и/или целых фраз). Мы обнаружили, что с уменьшением размера словаря распределение ранга-частотности все равно следует закону Ципфа даже на уровне токенов-подслов. По мере дальнейшего увеличения размера словаря для включения токенов, состоящих из нескольких слов и/или целых фраз, распределение ранга-частотности больше не следует "чистому" степенному закону, а скорее напоминает суперпозицию двух законов Ципфа, следуя концепции "согласованности" (coherence), предложенной в [11]. Когда размер словаря достигает определенного значения, возникает порог, в окрестности которого распределение ранга-частотности проходит фазовый переход. Параметр аппроксимирующего распределения Ципфа сдвигается. Мы предполагаем, что такое поведение вызвано тем, что в случае более

крупных словарей мы имеем дело с двумя подмножествами токенов, каждое из которых согласовано само по себе. На рисунке 3.2 демонстрируется "изгиб" закономерного распределения Ципфа.

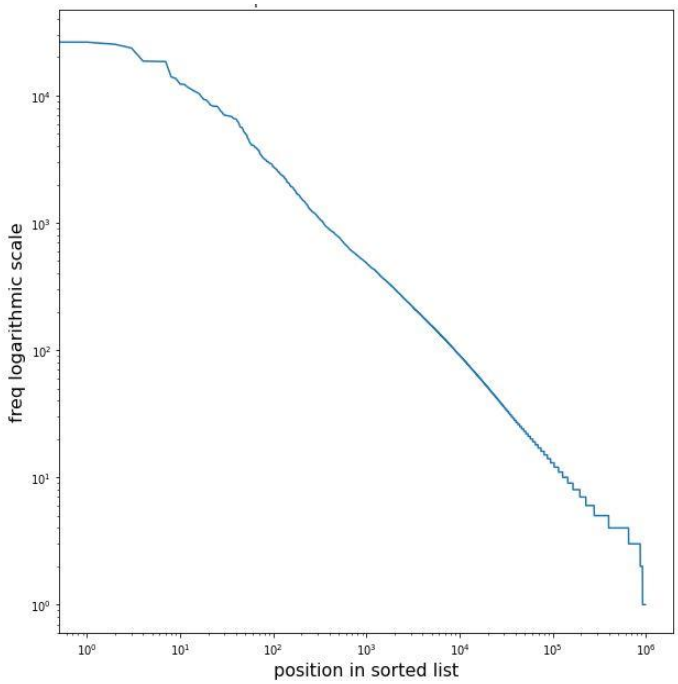


Рисунок 3.2: Диаграмма ранга-частотности для словаря размером 1 миллион демонстрируя "изгиб" закономерного распределения Ципфа.

Как показано в результатах экспериментов с различными размерами словаря на рисунках 3.3–3.6, распределение начинает вести себя как суперпозиция двух законов Ципфа в тот момент, когда длина токена превышает некоторый порог

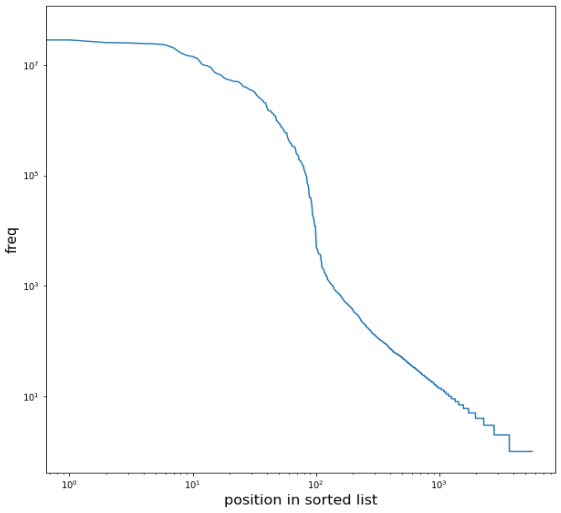


Рисунок 3.3: Размер словаря 5000 токенов

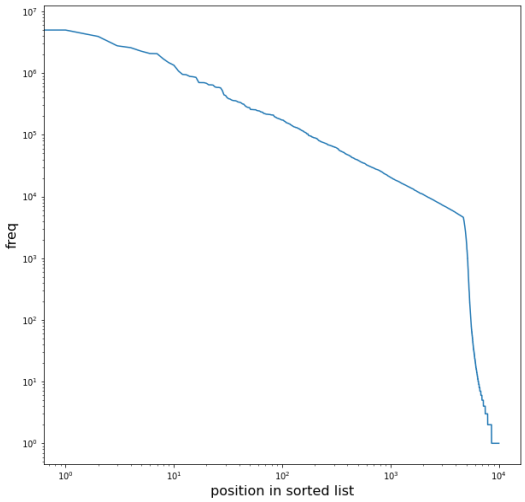


Рисунок 3.4: Размер словаря 10000 токенов

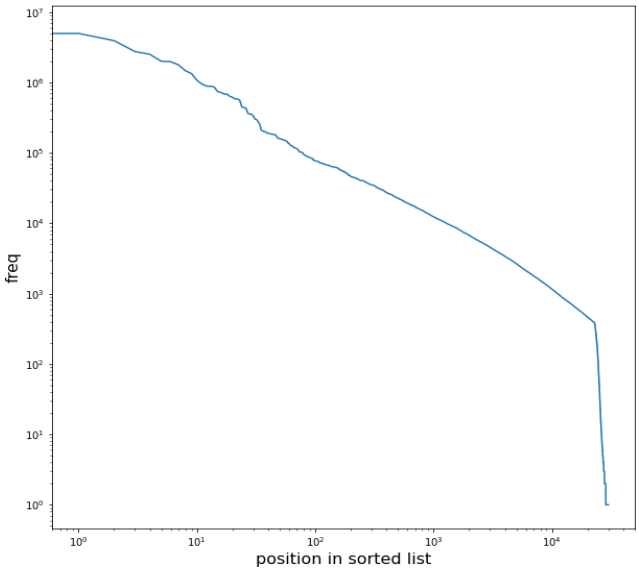


Рисунок 3.5: Размер словаря 30 000 токенов

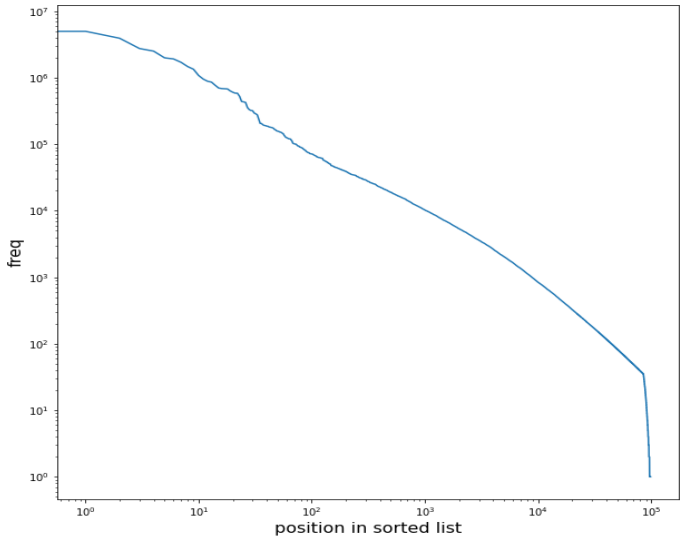


Рисунок 3.6: Размер словаря 100 000 токенов

Мы предполагаем, что этот порог определяется семантикой: поведение распределения в головной части таких распределений отличается от поведения в хвостовой части. Гипотеза состоит в том, что головная часть в основном состоит из более коротких токенов с возможными семантическими вариациями, тогда как хвостовая часть в основном состоит из более длинных токенов, связанных с одной конкретной семантической областью.

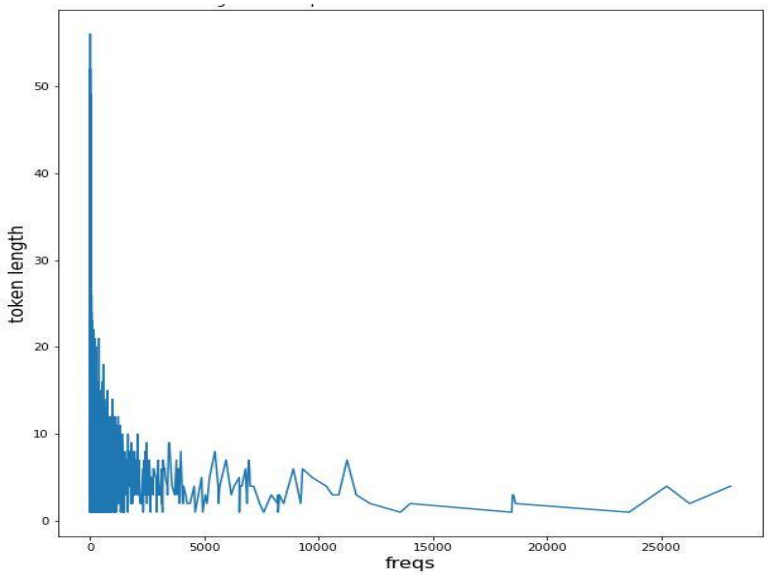


Рисунок 3.7: Распределение длин токенов

Как более короткие токены с несколькими значениями, так и более длинные токены с единственным значением являются согласованными и демонстрируют распределение в соответствии с законом Ципфа, если рассматривать их отдельно. Это видно на графиках распределения с более маленькими размерами словаря, которые, предположительно, состоят в основном из более коротких токенов (см. пример на рисунке 3.3). Вместе эти два подмножества уже не проявляют чистого поведения Ципфа. "Тяжелый хвост" такого распределения явно виден на рисунке 3.2. Рисунок 3.7 показывает, что хвост в основном состоит из более длинных токенов, которые, как показано ниже, имеют скорее одно значение, чем множество значений.

3.3. Эксперименты с семантикой

Для наглядного исследования качественных различий между этими двумя распределениями мы провели дополнительный эксперимент с двумя подмножествами токенов: одно из головной части распределения словаря с миллионом токенов и другое из хвостовой части. Как и ожидалось, подмножество "головы" состояло из более коротких токенов, в основном слов, а подмножество "хвоста" состояло из более длинных токенов, в

основном фраз и частей предложений. Поскольку в головной части, а также в средней части распределения нет длинных токенов (см. рисунок 3.7), и из-за характера эксперимента (нас в основном интересовало восприятие аудиторией более коротких токенов из головы и более длинных токенов из хвоста), мы отфильтровали несколько коротких токенов, которые могут встречаться в хвосте, и оставили в хвостовой части только длинные токены. Мы провели опрос среди группы профессиональных лингвистов относительно перемешанной последовательности токенов из двух подмножеств, чтобы выяснить разницу между этими подмножествами на основе мнения людей с соответствующим профессиональным опытом. Были заданы следующие вопросы относительно каждого отдельного токена X в наборе: 1) Можете ли вы переформулировать X? 2) Сколько значений имеет X в зависимости от контекста? 3) Можете ли вы поместить X в контекст?

Рисунки 3.8–3.9 иллюстрируют результаты одного из опросов: более короткие токены из головной части распределения характеризуются семантической неоднозначностью и часто могут иметь несколько значений в зависимости от контекста. Более длинные токены из хвостовой части распределения, как правило, имеют одно или два конкретных значения.

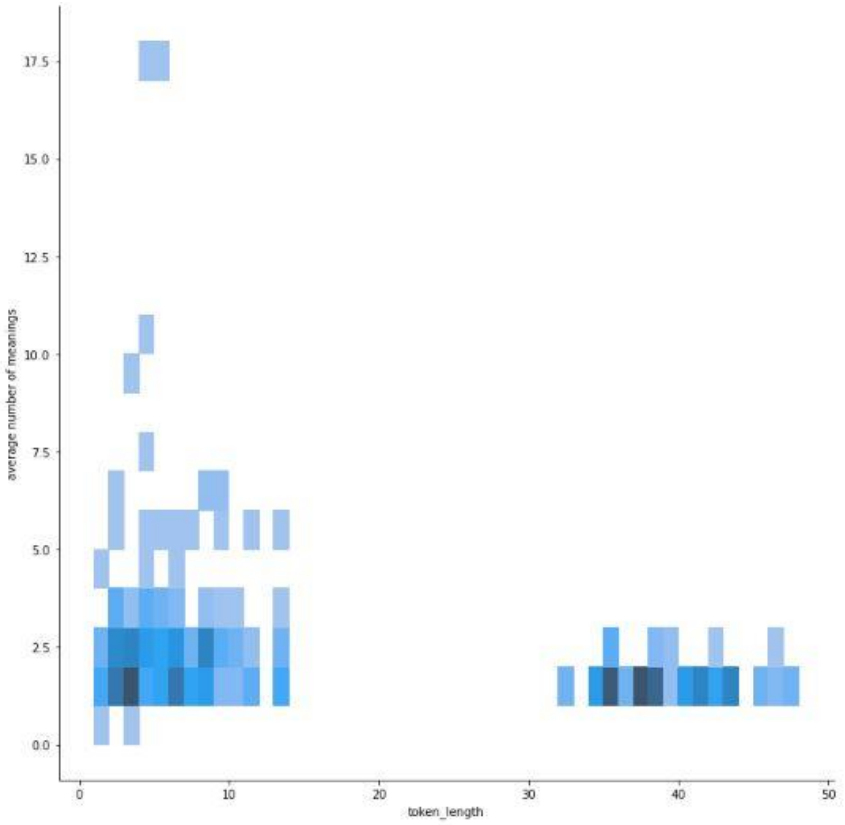


Рисунок 3.8: результаты опроса: длина токена vs среднее кол-во значений (хитмапа) аннотированных лингвистом

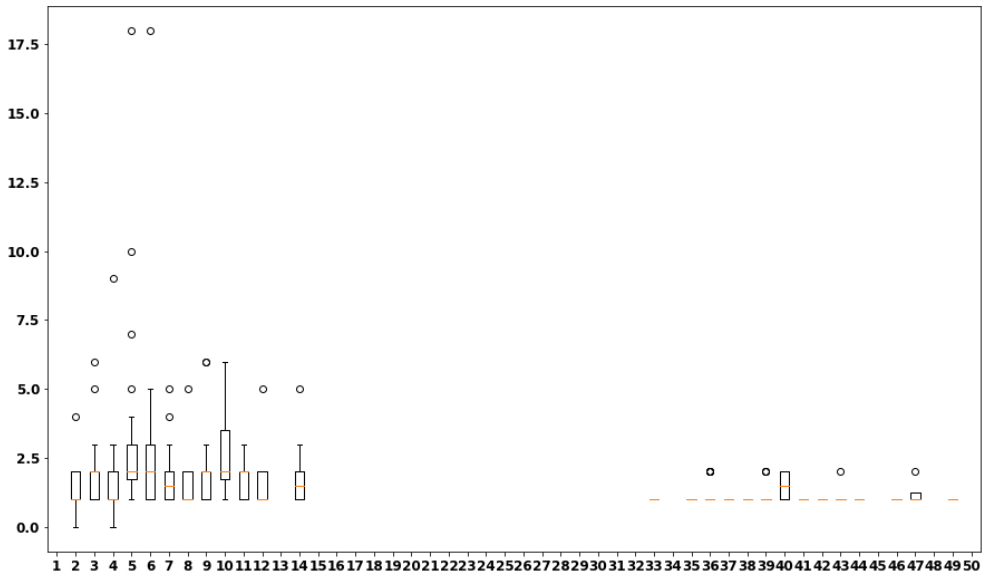


Рисунок 3.9: результаты опроса: длина токена vs среднее кол-во значений (боксплот + отклонения), аннотированных лингвистом

На рисунках 3.10–3.12 показано, что более короткие, семантически неоднозначные токены могут легко вписываться в различные контексты, в то время как более длинные токены с одним значением представляют собой контекст сами по себе: попытка вписать их в контекст приводит к изменению нескольких более коротких токенов, из которых они состоят, при этом остальное остается неизменным. Разница явно видна по нормализованному расстоянию Левенштейна между исходным токеном и токеном в контексте. Стоит также отметить, что распределение нормализованного расстояния Левенштейна в левой части диаграммы также визуально напоминает закон Ципфа — это может быть темой для более детального изучения с более широкой аудиторией.

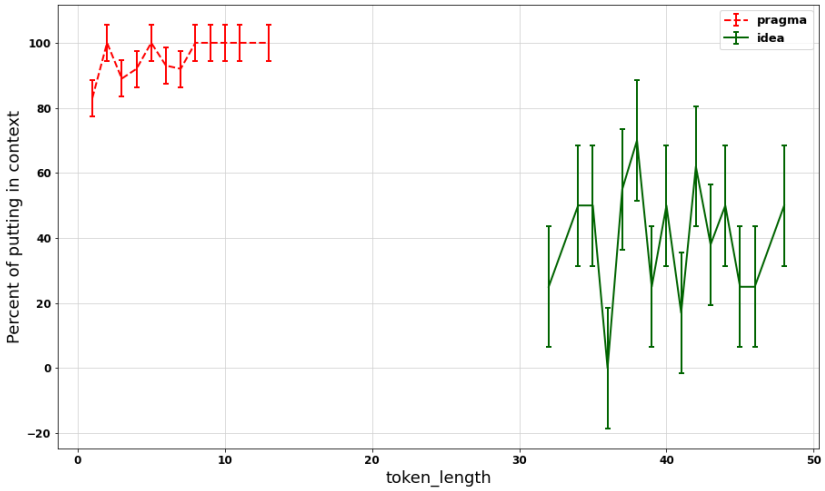


Рисунок 3.10: результаты опроса: длина токена vs процент помещения его в контекст

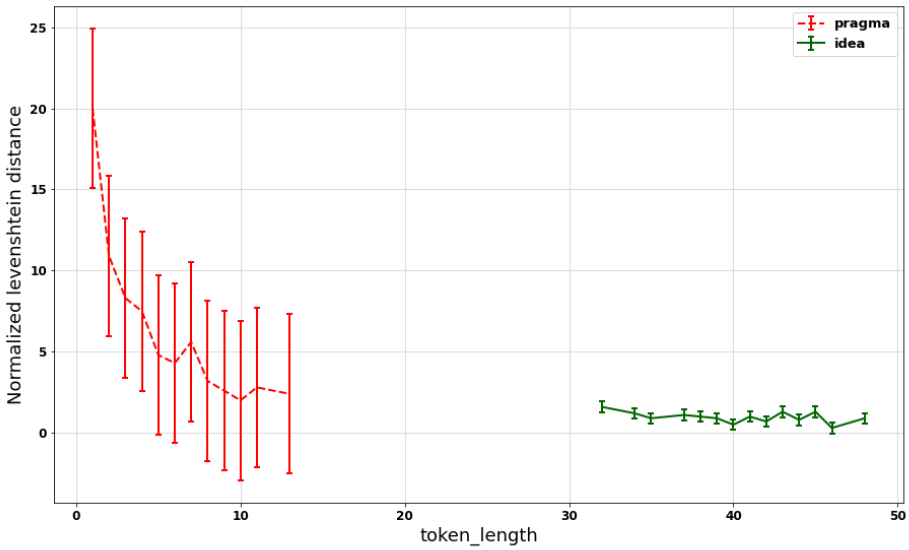


Рисунок 3.11: Результаты опроса: длина токена vs нормализованное расстояния Левенштейна между исходным токеном и токеном, помещенным в контекст. Токены из "головы" могут быть помещены в различные контексты, которые существенно отличаются друг от друга. Токены из "хвоста" обычно встраиваются в сходные контексты, сравнимые по размеру с самим токеном.

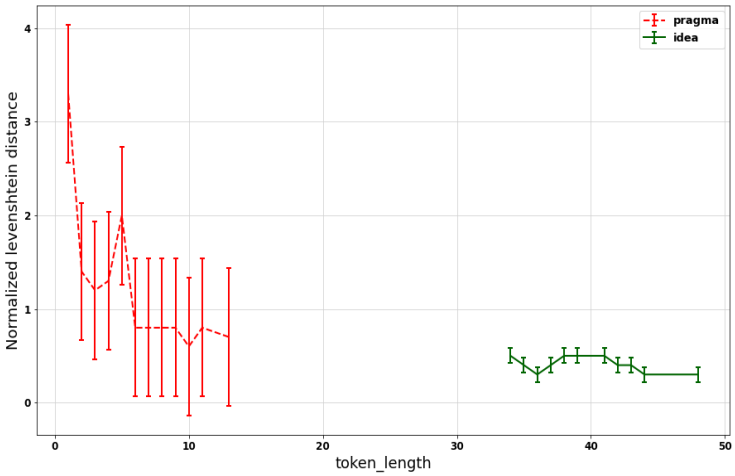


Рисунок 3.12: результаты опроса: длина токена vs нормализованное расстояние Левенштейна между исходным токеном и токеном после переформулировки

Результаты были сгруппированы в соответствии с тем, к какому подмножеству относится каждый токен, и показывают, что, исходя из мнения людей с профессиональным лингвистическим опытом:

- "Головные" токены легко заменяются синонимами, омонимами или фразами с синонимичным или омонимичным значением, полное содержание токена меняется при переформулировании, вместе с возможным изменением значения;
- "Хвостовые" токены сложно переформулировать, в основном путем изменения одного или нескольких более коротких токенов, из которых они состоят, при этом остальное остается неизменным. Независимо от переформулирования, значение всего токена не меняется;
- "Головные" токены часто имеют несколько значений, тогда как "Хвостовые" токены в основном имеют одно или два значения;
- "Головные" токены легко вписываются в контекст и могут иметь различные значения в зависимости от контекста;

- Вставка "хвостового" токена в контекст не так проста и производится путем вставки нескольких более коротких токенов в контекст более длинного токена, при этом значение всего длинного токена остается неизменным.

Эти результаты показывают, что "головные" и "хвостовые" токены отличаются как семантически, так и с точки зрения статистики. "Головные" токены имеют семантическую неоднозначность, их легко поместить в различные контексты с разными значениями. "Хвостовые" токены имеют однозначное соответствие с определенным семантическим концептом. Их трудно поместить в новый контекст, отличный от контекста, который они сами определяют. Хвостовая часть распределения ранг-частоты, в основном состоящая из таких токенов, демонстрирует поведение, присущее распределению ранг-частоты отдельных, когерентных наборов токенов. И "головные", и "хвостовые" подмножества являются когерентными с точки зрения [11], и распределение каждого из них, следует закону Ципфа. В отличие от этого, распределение их объединения больше напоминает суперпозицию двух законов Ципфа, связанных с двумя различными когерентными наборами токенов.

3.4. Типология

Мы предлагаем ввести следующие термины, которые характеризуют различные типы токенов:

Атом (Atom) - самый маленький элемент дискретной последовательности, который нельзя разделить на более мелкие части. В письменном языке атомами считаются отдельные символы. Исходный набор атомов считается исчерпывающим списком ограниченного размера.

Прагма (Pragma) - часть дискретной последовательности, состоящая из атомов и представляющая собой целостную часть идеи. Прагма имеет семантическую неоднозначность и может иметь разные значения в

зависимости от контекста. В письменном языке прагмами являются символьные n-граммы, слова и словесные n-граммы.

Идея (Idea) - часть дискретной последовательности, состоящая из прагм и имеющая однозначное соответствие с определенным концептом (семантическим концептом в случае языка). В письменном языке идеи могут быть представлены семантически отличными предложениями, устойчивыми фразами и коллоквиализмами с определенными значениями.

Мы не концентрируемся на атомах в данной работе, но предлагаем их ввести для полноты. Мы считаем, что такая структура атом-прагма-идея помогает понять и описать некоторые концепции, связанные как со статистическим поведением, так и с семантикой определенных наборов токенов. Мы также полагаем, что такая трехуровневая структура может найти применение в исследованиях обработки дискретных последовательностей, связанных с многими областями, кроме естественного языка: химия, биология, музыка и т. д.

Языковые модели обычно используют размер словаря, меньший, чем тот, который мы использовали в экспериментах. Поэтому они представляют идеи как последовательности прагм, вместо того чтобы формировать отдельный, согласованный поднабор токенов из идей. Это затрудняет определение точки "сдвига фазы", когда более длинная последовательность прагм становится идеей и начинает проявлять однозначное соответствие с определенным семантическим концептом. Мы считаем, что это может быть одной из причин, почему языковые модели показывают низкую производительность при работе с более длинными последовательностями, в отличие от людей, у которых таких проблем нет. Мы предлагаем назвать этот феномен "прагматическим ограничением" - способность статистического обучения работать с прагматической токенизацией, которая не использует сокращение семантической неоднозначности, характерной для токенов типа идея.

3.5. Автоматическая оценка единиц в рамках типологии

Статистический анализ и лингвистическая оценка позволили сформировать разноплановые группы единиц текста. Тем не менее восприятие свойств, по которым мы различаем элементы в рамках типологии у языковых моделей, отличается от человеческого. В качестве демонстрации этого феномена необходимо было разработать и имплементировать методологию автоматической оценки токенов на принадлежность к тому или иному классу на базе языковой модели. Предложенная экспертам (см. раздел 3.3) методология фактически является аналогом двунаправленной текстовой генерации, поэтому при создании алгоритма оценивания токенов на разделение внутри типологии решено было взять эту задачу за основу. Важно подчеркнуть, что в предлагаемых экспертам вопросах требовалось окружить единицу контекстом с обеих сторон, а не продолжить фразу, используя указанный элемент как префикс.

Известные нам большие трансформерные языковые модели достаточно неплохо справляются с текстовой генерацией, т.е. задачей предсказания следующего вероятного слова на основе предыдущей последовательности слов. Основной линейкой таких текстовых генераторов являются GPT модели. Однако GPT-3 [3], несмотря на внушительное увеличение как количества параметров (более чем в 100 раз по сравнению с GPT-2), так и количества данных для обучения, обладает существенным недостатком: доступ к исходному коду закрыт, есть доступ только к API, причем тарифицируемый, что не позволяет решить задачу. GPT-2 же имеет открытый исходный код и вполне неплохо справляется с задачей генерации, но в направлении продолжения префикса, иными словами, слева направо. При этом известна проблема зацикливания GPT-2 при генерации длинных последовательностей. BERT [15] же подобные модели в принципе хуже адаптируются под генерацию чем GPT и страдают от шумов. Хорошей альтернативой является XLNET [50]: 1) Размер корпусов на которых

обучалась модель значительно превосходит аналогичные для GPT-2 и BERT; 2) Код находится в открытом доступе; 3) Внутри себя XLNET использует SentencePiece [25] в качестве токенизатора, который более гибок в выборе методологий разделения на подслова, лучше справляется с редкими и неизвестными словами и обладает большей сжимаемостью; 4) Архитектурно лучше адаптирован под генерацию в два направления. Разберем этот пункт подробнее.

XLNET уникален тем, что использует все перестановки входной последовательности, таким образом он может генерировать как однонаправленно (причем и слева направо, и справа налево), так и двунаправленно. При предсказании целевого слова в последовательности контекстные слова, к которым модель имеет доступ, определяются факторизационным порядком. Для иллюстрации представим, что у нас есть последовательность $x = [x1, x2, x3, x4]$. Один из возможных факторизационных порядков: x3→x2→x4→x1. При предсказании целевого слова x4 модель имеет доступ только к контекстным словам {x3, x2}; если целевым словом является x2, она видит только {x3}. На практике целевое слово устанавливается последними несколькими словами в факторизационном порядке (например, x4 и x1), поэтому модель всегда видит некоторые контекстные слова для предсказания. Еще одной архитектурной особенностью является то, что используется рекурентный механизм сегмента [14]. Этот механизм вдохновлен усеченным обратным распространением во времени, используемым для обучения рекуррентных нейронных сетей, где начальное состояние последовательности инициализируется конечным состоянием предыдущей последовательности. Механизм сегментного повтора работает аналогичным образом, кэшируя скрытые состояния блоков трансформера из предыдущей последовательности и позволяя текущей последовательности использовать их в процессе обучения. Это позволяет XLNET моделировать долгосрочные

зависимости за пределами его максимальной длины последовательности. Все это явилось хорошими аргументами для использования XLNET в качестве модели двунаправленной генерации.

В целом, чтобы сделать предсказание в одну сторону, нам нужно передать модели токенизированный текст, индексы замаскированных слов и маски перестановки. Маски перестановки необходимы для отключения входных токенов от взаимодействия с замаскированными токенами. Разовьем эту идею и сделаем генератор двунаправленным и избавимся от зацикливаний. Еще один элемент, который необходимо добавить это выборку из top-k токенов: на каждой итерации модель будет предсказывать top-k токенов для замаскированного слова справа или слева от начальной фразы. Затем мы добавим случайный токен из top-k в начальную фразу и повторим итерацию n раз. Далее, чтобы побороть проблему зацикливаний и для повышения связности генерируемого текста, необходимо использовать beam search: мы можем увеличить вероятность нахождения связанных последовательностей слов, генерируя не только по одному слову с каждой стороны начальной фразы, но создавая определенное количество пучков (beams) последовательностей слов и выбирая один из наиболее вероятных пучков заданной длины. Таким образом, весь алгоритм выглядит как: 1) На первом этапе берем начальную фразу и генерируем справа от нее определенное количество пучков (beams) определенной длины в направлении слева направо (на каждом этапе поиска пучков мы выбираем следующие кандидаты на токены с помощью top-k сэмплирования) 2) На втором этапе мы выбираем случайный пучок из top-k наиболее вероятных пучков и добавляем его к начальной фразе 3) Полученная новая фраза служит в качестве начальной фразы для третьего шага, в котором мы генерируем определенное количество пучков справа от новой начальной фразы 4) На четвертом этапе мы выбираем случайный пучок из top-k пучков, полученных на третьем шаге, и добавляем этот пучок к новой

начальной фразе. Полученная фраза служит отправной точкой для следующей итерации. Также вариация в сторону увеличения параметра температуры позволяет снизить уверенность модели в выборе наиболее вероятных токенов. Это позволяет сделать генерацию более разнообразной и избежать застревания с наиболее вероятными повторяющимися последовательностями токенов.

Далее встает вопрос оценки того, что сгенерировано моделью. Необходимо было воспроизвести механизм принятия человеком решения о помещении единицы в контекст, его уверенности относительно того, что единица однозначно помещается в один и тот же контекст, или обратно - наличие разноплановых контекстов. При выдаче задания человеку мы опирались на то, что все люди (особенно в экспертной области) разделяют какое-то общее понимание и знание о языке и действительно обнаружили феномен того, что абсолютно разные люди помещали в один и тот же контекст определенные токены. Таким образом, чтобы достичь статистической значимости полученных результатов, нам тоже потребовалось для каждого токена осуществлять повторную генерацию контекста. Тем не менее, улучшив качество генерации с помощью техник пучкового поиска и top-k сэмплирования, мы оказываемся в ситуации (в силу множественной генерации) более искусственно разнообразного контекста. Однако механизм оценки качества сгенерированного материала через уверенность все же имеется: это метрика perplexity (перплексия).

$$2^{-\frac{1}{N}\sum_{i=1}^{N}\log P(\text{word}_i|\text{word}_1,\text{word2},\ldots,\text{word}_{i-1})}$$

Чем меньше значение перплексии, тем больше модель уверена.

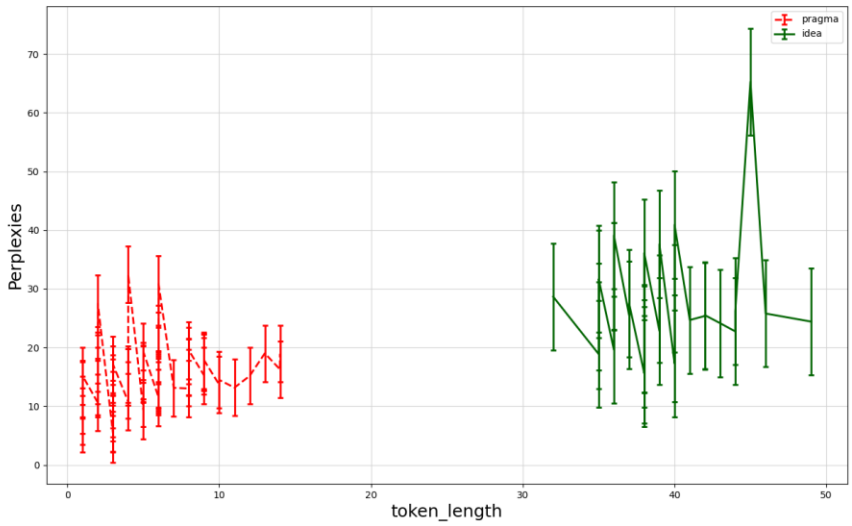


Рисунок 3.13: Разделимость классов pragma / idea по перплексии для той же выборки, что была предложена экспертам.

Из приведенных графиков видно, что большая уверенность модели достигается при работе с pragma, а не с idea, что противоположно тому, как человек воспринимает единицы разных классов: больше уверенности и однозначности при работе с idea нежели с pragma. Это наблюдение согласуется с анализом феномена прагматических ограничений, приведенном в конце раздела 3.4.

Тем не менее перед нами стояла задача проклассифицировать большее количество единиц. Конкретнее, хочется оценить, насколько сбалансирован относительно наполняемостью pragma / idea словарь классической языковой модели. Словари уже упомянутых языковых моделей в зависимости от конфигурации (base / large) составляют от ~20K до ~30K токенов. Описанная выше методология автоматической оценки все-таки требует значительных вычислительных ресурсов даже при наличии графических процессоров на кластере, поэтому в силу необходимости получить результат за разумное время (несколько дней вместо недель) было принято решение отделить часть токенов из словаря,

провести над ними оценку автоматическим методом, а по полученной от модели разметки обучить классификатор и проклассифицировать им оставшийся словарь. Чтобы лучше репрезентировать представленные типы токенов при отделении части, возьмем равномерное распределение по корзинам (bin) и в них самый центральный элемент. Далее, применив методологию автоматической оценки, получаем распределение pragma vs. idea в отделенной части словаря. Изучив распределение значений перплексии, мы пришли к выводу, что можно отсекать по нижней квартили, таким образом получая разметку для бинарной классификации. Далее, так как в рамках типологии pragma и idea различаются статистически и семантически, предположительно, что в свою очередь модель извлекает эти различия из представления (эмбеддинга) токена, так как языковые модели оперируют именно ими. Соответственно, по эмбеддингам и имеющейся разметке построен классификатор. Мы попробовали различные подходы от байесовских классификаторов [28], SVM [9], KNN [10] до Random Forest [2] и XGBoost [7], последние два показали сопоставимое между собой качество и сильно выше чем остальные подходы. Результаты представлены в таблице 3.1.

Таблица 3.1: Результаты работы различных классификаторов по F-мере для каждого класса из типологии

Подход	F-мера класс pragma	F-мера класс idea
Gaussian NB	0.43	0.57
Bernoulli NB	0.60	0.49
KNN	0.66	0.56
SVM	0.80	0.76
XGBoost	0.82	0.76
Random Forest	0.82	0.80

При классификации обученным классификатором оставшейся части выборки, получено примерно такое же (с поправкой на не 100% качество самого классификатора) распределение по классам, что и у данных для обучения, которые выбирались равномерно из всей выборки. Это позволяет сделать вывод о том, что вся методология автоматической оценки даже при небольшом элементе приближенности в последнем шаге всего пайплайна достаточно хорошо репрезентирует наполняемость словаря языковой модели pragma и idea.

3.6. Выводы

- Были обнаружены как статистические, так и семантические различия для двух отдельных подмножеств токенов, выявляемых при увеличении размера словаря любого токенизационного алгоритма в рамках закона Ципфа
- Была разработана новая типология гранулярности языковых единиц
- Справедливость и применимость типологии была проверена как в рамках лингвистического исследования среди группы профессионалов, так и с помощью подхода на базе языковой модели
- Для валидации типологии был разработан фреймворк автоматической оценки токена на принадлежность к классу и проверена его применимость на реальном словаре языковой модели

Заключение

В результате проведенного исследования выработана легковесная методология выделения крупных единиц гранулярности текста на базе BERT, предобученного под NSP, и проведено сопоставление с более архитектурно сложными подходами, по результатам которого выявлено, что предложенный способ не сильно хуже по качеству, но требует меньше вычислительных ресурсов. Также на основе экспериментов с различными алгоритмами токенизации и выявленными статистическими и семантическими особенностями когерентных подгрупп токенов в рамках закона Ципфа предложена новая типология текстовых единиц. Разработана методология автоматической разделимости текстовых единиц по типам и применена к активному словарю языковой модели. Таким образом, глобально выработан новый семантически ориентированный подход к дискретизации текста на разных уровнях и проверена его применимость с помощью языковых моделей.

Дальнейшее развитие в данной области может вестись по следующим направлениям.

Во-первых, можно провести дополнительные исследования по соотнесению совсем крупных единиц текста, описанных в главе 2 с выработанной типологией. Также можно развить идею сегментации с неполным дообучением, дистилляцией модели. Продуктивным также представляется дальнейшее исследование выделенных в этой работе сдвигов в метриках по задаче сегментации и разработка более сбалансированной.

Во-вторых, можно расширить применимость разработанных типологии и методологии автоматической оценки и выработать новый подход к токенизации, балансирующий единицы типов pragma и idea в словаре и проверить, как это влияет на показатели моделей на конечных задачах.

Список литературы

[1] Beeferman, D., Berger, A., & Lafferty, J.. (1999). Statistical models for text segmentation. *Machine learning 34(1):177–210.*

[2] Breiman, L. (2001). Random forests. *Machine learning, 45(1), 5-32.*

[3] Brown, T., Mann, B. F., Ryder, N. C., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J. C., Winter, C., . . . Amodei, D. (2020). Language Models are Few-Shot Learners. In *Neural Information Processing Systems* (Vol. 33, pp. 1877–1901).

[4] Cancho, R. F., & Solé, R.V. (2001). Two Regimes in the Frequency of Words and the Origins of Complex Lexicons: Zipf's Law Revisited. *Journal of Quantitative Linguistics, 8,* 165-173. doi:10.1076/jqul.8.3.165.4101

[5] Cancho, R. F., & Vitevitch, M.S. (2018). The Origins of Zipf’s meaning-frequency Law. *Journal of the Association for Information Science and Technology, 69(11):*1369–1379.

[6] Carlson L, Marcu D, Okurowski M E. Building a discourse-tagged corpus in the framework of rhetorical structure theory. *Springer, 2003*

[7] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 785-794).*

[8] Clark, K., Luong, M.-T., Le, Q.V., & Manning. C.D. (2020). ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. *arXiv preprint arXiv:2003.10555*.

[9] Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine learning, 20(3), 273-297.*

[10] Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory, 13(1), 21-27.*

[11] Cristelli, M., Batty, M., & Pietronero, L. (2012). There is More than a Power Law in Zipf. *Scientific reports, 2:812.*

[12] Crowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H.W., … Fiedel, N. (2022). PaLM: Scaling Language Models with Pathways. *arXiv preprint arXiv: 2204.02311.*

[13] Dai, Z., & Huang, R. (2019). A Regularization Approach for Incorporating Event Knowledge and Coreference Relations into Neural Discourse Parsing. https://doi.org/10.18653/v1/d19-1295

[14] Dai, Z., Yang, Z., Yang, Y., Carbonell, J.G., Le, Q.V., & Salakhutdinov, R. (2019). Transformer-XL: Attentive Language Models beyond a Fixed-Length Context. *ArXiv, abs/1901.02860*.

[15] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805.*

[16] Gage, P. (1994). A new algorithm for data compression. *C Users J., 12(2):*23–38

[17] Harris, Z. (1954). Distributional structure. *Word, 10(23):*146–162.

[18] He, P., Liu, X., Gao, J., & Chen, W. (2021b). DEBERTA: DECODING-ENHANCED BERT WITH DISENTANGLED ATTENTION. In *International Conference on Learning Representations*.

[19] Hochreiter, S. & and Schmidhuber, J. (1997) Long short-term memory. *Neural computation 9(8):1735–1780*

[20] Jiang, F., Fan, Y., Chu, X., Li, P., Zhu, Q., & Kong, F. (2021). Hierarchical Macro Discourse Parsing Based on Topic Segmentation. *Proceedings of the AAAI Conference on Artificial Intelligence*, *35*(14), 13152-13160.

[21] Kishimoto, Y., Murawaki, Y., & Kurohashi, S. (2020). Adapting BERT to Implicit Discourse Relation Classification with a Focus on Discourse Connectives. In *Language Resources and Evaluation* (pp. 1152–1158).

[22] Kobayashi, N., Hirao, T., Kamigaito, H., Okumura, M., & Nagata, M. (2020). Top-Down RST Parsing Utilizing Granularity Levels in Documents. *Proceedings of the AAAI Conference on Artificial Intelligence*,*34*(05), 8099–8106.

[23] Koshorek, O., Cohen, A., Mor, N., Rotman, M. & Berant, J. (2018). Text Segmentation as a Supervised Learning Task. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 469–473.

[24] Kudo, T. (2018). Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates. *arXiv preprint arXiv: 1804.10959.*Начало формы

[25] Kudo, T. & Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing. *arXiv preprint arXiv: 1808.06226.*

[26] Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. *arXiv preprint arXiv: 1909.11942.*

[27] Le, Q. H., Sicilia-Garcia, E. I., Ming, J., & Smith, F. J. (2002). Extension of Zipf's Law to Words and Phrases. *In Proceedings of the 19th International Conference on Computational Linguistics, 1,* 1-6. doi:10.3115/1072228.1072345.

[28] Lewis, D. D. (1998). Naive (Bayes) at forty: The independence assumption in information retrieval. *In European conference on machine learning (pp. 4-15). Springer.*

[29] Li, J., Li, M., Qin, B., & Liu, T. (2022). A survey of discourse parsing. *Frontiers of Computer Science*, *16*(5). https://doi.org/10.1007/s11704-021-0500-z

[30] Lukasik, M., Dadachev, B., Papineni, K & Simões, G. (2020). Text Segmentation by Cross Segment Attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4707–4716.

[31] Mann W C, Thompson S A. (1988). Rhetorical structure theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse, 8(3): 243–281*

[32] Meister, C., Pimentel, T., Haller, P., Jäger, L.,Cotterell, R. & Levy, R. (2021). Revisiting the Uniform Information Density Hypothesis. *In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 963–980.*

[33] Nikkarinen, I., Pimentel, T., Blasi, D., & Cotterell, R. (2021). Modeling the unigram distribution. *In Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021,* 3721–3729.

[34] Papineni, K., Roukos, S., Ward, T., & Zhu, W. (2002). Bleu: a method for automatic evaluation of machine translation. *In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, pages 311–318*

[35] Pimentel, T., Nikkarinen, I., Mahowald, K., Cotterell, R., & Blasi, D. (2021). How (non-) optimal is the lexicon? *In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 4426–4438.

[36] Pimentel, T., Roark, B., Wichmann, S., Cotterell, R., & Blasi, D. (2021). Finding Concept-specific Biases in Form–Meaning Associations. *In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* 4416-4425. doi: 10.18653/v1/2021.naacl-main.349

[37] Prasad R, Dinesh N, Lee A, Miltsakaki E, Robaldo L, Joshi A K, Webber B L. The penn discourse treebank 2.0. *In: LREC. 2008*

[38] Provilkov, I., Emelianenko, D. & Voita, E. (2020). BPE-Dropout: Simple and Effective Subword Regularization. *arXiv preprint arXiv:1910.13267.*

[39] Rutherford, A., & Xue, N. (2015). Improving the Inference of Implicit Discourse Relations via Classifying Explicit Discourse Connectives. *In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 799–808, Denver, Colorado.*

[40] Rutherford, A., Vera Demberg, V., & Xue, N. (2017). A systematic study of neural discourse models for implicit discourse relation. I*n Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics:* Volume 1, Long Papers, volume 1, pages 281–291.

[41] Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108.*

[42] Sennrich, R., Haddow, B., & Birch, A. (2016). Neural machine translation of rare words with subword units. I*n Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1715– 1725.*

[43] Shi, W., & Demberg, V. (2019). Next Sentence Prediction helps Implicit Discourse Relation Classification within and across Domains. https://doi.org/10.18653/v1/d19-1586

[44] Takahashi, S. & Tanaka-Ishii, K. (2017). Do neural nets learn statistical laws behind natural language? *PloS one, 12(12),* e0189326.

[45] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is All You Need. *Neural Information Processing Systems*, *30*, 5998–6008.

[46] Wang, Y., Li, S., & Yang, J. (2018). Toward fast and accurate neural discourse segmentation. *In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 962–967.*

[47] Wu, Y., Schuster, M., Chen, M.Z., Le, Q.V., Norouzi, M., Macherey, W., ... Macherey, K. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144.*

[48] Xing, L., Hackinen, B., Carenini, G. & Trebbi, F. (2020). Improving Context Modeling in Neural Topic Segmentation. *AACL.*

[49] Yang, C. (2013). Who's afraid of george Kingsley zipf? or: Do children and chimps have language? *Significance, 10(6):*29–34.

[50] Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., & Le, Q.V. (2019). XLNet: Generalized Autoregressive Pretraining for Language Understanding. *arXiv preprint arXiv:1906.08237.*

[51] Zhang, H. (2008). Exploring regularity in source code: Software science and zipf's law. *15th Working Conference on Reverse Engineering*, 101–110. IEEE.

[52] Zipf, G.K. (1932). *Selected Studies of the Principle of Relative Frequency in Language.* Cambridge, Massachusetts: Harvard University Press.