ИПМ им.М.В.Келдыша РАН • Электронная библиотека

Препринты ИПМ • Препринт № 67 за 2022 г.

ISSN 2071-2898 (Print)
ISSN 2071-2901 (Online)

М.Ю. Кислицына

Анализ влияния предобработки текстов на идентификацию авторов методом биграмм

Рекомендуемая форма библиографической ссылки: Кислицына М.Ю. Анализ влияния предобработки текстов на идентификацию авторов методом биграмм // Препринты ИПМ им. М.В.Келдыша. 2022. № 67. 18 с. https://doi.org/10.20948/prepr-2022-67
https://library.keldysh.ru/preprint.asp?id=2022-67

О р д е н а Л е н и н а
ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ
имени М.В.Келдыша
Р о с с и й с к о й а к а д е м и и н а у к

М.Ю. Кислицына

Анализ влияния предобработки текстов на идентификацию авторов методом биграмм

Москва – 2022

Кислицына М.Ю.

Анализ влияния предобработки текстов на идентификацию авторов методом биграмм

На примере достаточно представительного корпуса авторов и текстов проведен сравнительный анализ влияния программ предобработки текстов на возможность идентификации авторов. Исследован вопрос чувствительности ошибки идентификации по доле изменения исходного текста. Показано, что авторское своеобразие сохраняется после предобработки практически на уровне оригинального текста.

Ключевые слова: машинная классификация, предобработка текстов, распределение биграмм, идентификация автора

Kislitsyna M.Yu.

The text preprocessing influence analyze for author identification problem by bigram method

On the example of sufficiently representative number of authors and texts, a comparative analysis of the impact of text preprocessing programs on the possibility of identifying authors is carried out. The question of the sensitivity of the identification error by the proportion of changes in the source text is investigated. It is shown that the author's originality is preserved after preprocessing almost at the level of the original text.

Keywords: machine classification, text preprocessing, bigram distribution, author identification

Содержание

Введение 3
1. Паттерны биграмм и идентификация автора 4
2. Анализируемые программы предобработки текстов 6
3. Результаты численных экспериментов распознавания автора 10
Заключение 16
Литература 17

Введение

Машинная обработка текстов на естественных языках (NLP, Natural Language Processing) представляет собой одно из важных направлений применения технологий искусственного интеллекта в задачах классификации текстов и массивов документов [1-3].

При автоматической обработке текстов для классификации и определения их атрибутов используются различные программы с целью предварительного преобразования исходной текстовой информации. В результате такого преобразования исходный текст меняется. Поскольку дальнейшие действия по классификации производятся с обработанным текстом, а выводы делаются относительно первоначального материала, возникает вопрос: насколько измененный текст сохраняет те свойства исходного, относительно которых и проводится анализ? Поскольку эффективность и адекватность различных методов предобработки зависят от целевой задачи, то абстрактное сравнение методов не особенно информативно, что отмечалось в работе [4]. Требуется их сравнение применительно к конкретной проблеме. Такой проблемой в данном исследовании выступает задача распознавания автора текста по эталонным распределениям частот буквосочетаний. Метод сравнения выборки с эталоном является стандартным приемом нахождения наиболее вероятного (ближайшего) соседа. Для задачи распознавания автора этот метод был сформулирован в работах [5-7]. Текст отдельного автора трансформировался в совокупность пар буквосочетаний. Средневзвешенное распределение частот по всем произведениям автора представляло собой эталон, с которым и сравнивались различные тексты.

Для идентификации атрибутов текстов на естественных языках (авторства, тематики, языка, жанра, эпохи написания) метод анализа текста существенно зависит от выбора атрибутов. Для определения жанра и эпохи эффективны семантические методы [8], а для определения автора и тематического направления – методы на основе анализа *n*-грамм [6, 7].

Объектами, изучаемыми в настоящей работе, являются программы предобработки текстов. Целевая функция анализа – составить эталон эмпирических частот двухбуквенных сочетаний в литературных текстах, написанных профессиональными писателями на русском языке или переведенными на русский язык. Это исследование основано на статистическом эксперименте с корпусом текстов на русском языке, результаты которого были описаны в [9].

Идея статистического распознавания атрибутов текстов состоит в том, что, предположительно, существует некоторая генеральная совокупность частот символов, которая отвечает данному автору и данному тематическому направлению, а каждый конечный текст реализуется как выборка из этой совокупности.

Близость между текстами понимается как близость между эмпирическими распределениями частот (ЭРЧ) в смысле нормы в L1. Автором неизвестного текста назначается тот, к эталону которого тестируемый текст находится ближе всего.

Ограничения точности такого метода распознавания связаны с тем, что некоторая часть биграмм обусловлена не авторским своеобразием, а общими свойствами языка. В то же время фильтрация, например, стоп-слов или неких общезначимых выражений может привести к искажению именно авторского своеобразия, поскольку писатель волен использовать последовательность слов в присущей только ему манере. То есть ошибки распознавания могут быть связаны как с анализом неотфильтрованного текста, так и с тем, что какая-то важная часть информации была ошибочно удалена в результате фильтрации.

В этой связи представляет определенный лингвистический интерес анализ точности распознавания автора внутри некоторого корпуса текстов в зависимости от способов предобработки этих текстов. В частности, насколько критичны для распознавания автора текста окончания и служебные слова в контексте рассматриваемого метода идентификации.

Далее в работе сравниваются результаты распознавания автора полного, то есть непредобработанного текста, с аналогичными результатами, полученными после применения различных программ.

1. Паттерны биграмм и идентификация автора

Пусть $D_a^i(j)$ – распределение частот символа j в i-м тексте автора a. Под символом будем подразумевать биграмму, то есть пару идущих подряд буквосочетаний без учета пробелов. Пусть также N_a^i есть количество указанных символов в данном i-м тексте. Паттерном (или эталоном) для идентификации автора текста будем называть взвешенное эмпирическое распределение частот по всей совокупности произведений данного автора, принадлежащих обучающему корпусу текстов, экспертно отнесенных к определенной тематической категории:

$$F_a(j) = \frac{1}{N_a} \sum_i N_a^i D_a^i(j), \quad N_a = \sum_i N_a^i . \tag{1}$$

На данном этапе считаем, что существует библиотека текстов, авторы которых достоверно известны. Суммирование по i в формуле (1) проводится от 1 до n_a – количества различных полных текстов (статей или книг) автора a.

Идентификация автора некоторого неизвестного, т.е. отсутствующего в библиотеке (обучающем корпусе), текста, проводится на основе принципа ближайшего эталона. Это означает, что авторство неизвестного текста с ЭРЧ $D(j)$ определяется формулой

$$a = \arg\min l(b), \quad l(b) = \sum_j \left| F_b(j) - D(j) \right| . \tag{2}$$

Расстояние между текстом и эталоном вычисляется по формуле

$$z_{ab}^{i} = \frac{1}{1-\delta_{ab}N_b^i / N_b}\sum_{j=1}^{J}\left|D_a^i(j) - F_b(j)\right|, \quad (3)$$

где δ_{ab} есть символ Кронекера. Формально для безошибочного распознавания автора текста требуется, чтобы

$$\forall i,a,b:\ z_{aa}^{i} < z_{ab}^{i},\ b \neq a. \quad (4)$$

Следует отметить, что авторские эталоны различаются между собой весьма незначительно. Если автор пишет на разные темы, то его эталон близок к частотам буквосочетаний общего русского лексикона. Тем не менее авторы отдельных текстов примерно в 90 % случаев идентифицируются верно. Ниже на Рис. 1 приведены эталоны биграмм для ста авторов корпуса [9]. Из совокупности этих графиков видно, что все они весьма близки между собой, а индивидуальные отличия незначительны. И если авторы полных текстов достаточно успешно распознаются по методу (2), то предобработка текстов может существенно изменить распределения частот. С какой точностью тогда будут определяться авторы? Эта проблема и рассматривается далее.

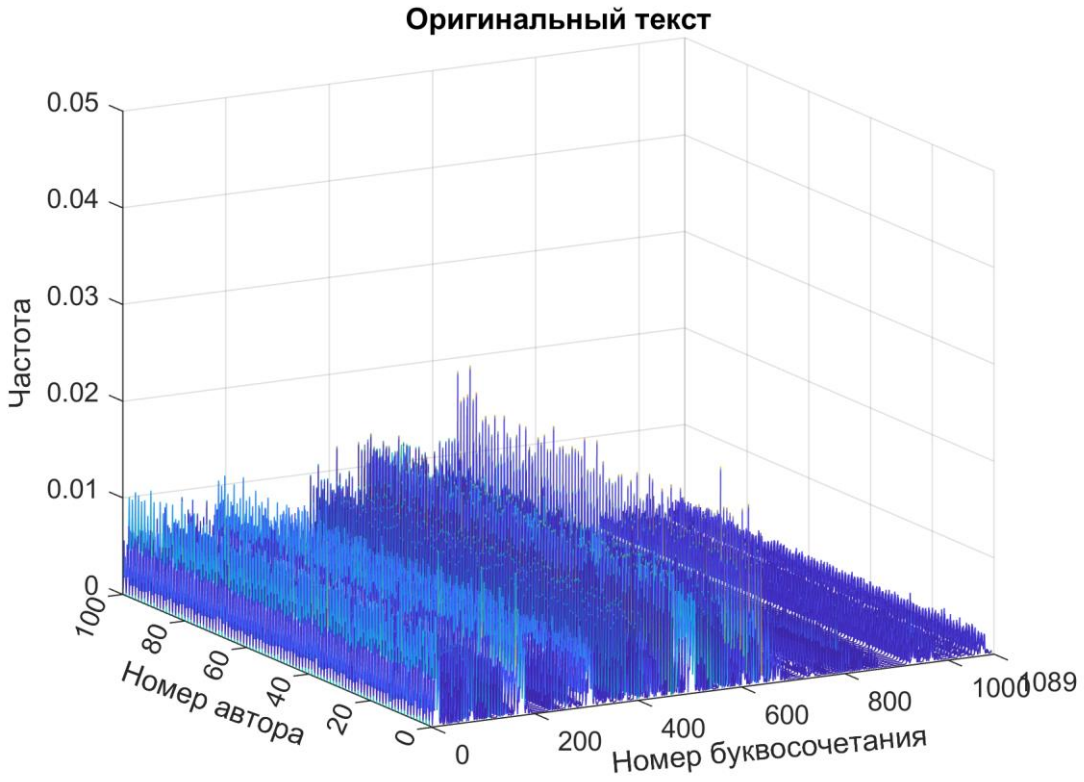

Рис. 1 – ЭРЧ эталонных биграмм авторов

Наиболее часто встречающиеся пары буквосочетаний имеют довольно малые величины эмпирических частот – порядка нескольких сотых. Чаще всего встречается сочетание «то», его частота составляет величину порядка 0,015.

С частотой около 0,01 встречаются сочетания «на», «ст», «но», «по», «не», «ен» и «ов». Затем идут около 250 биграмм с частотой несколько

тысячных. Остальные биграммы имеют частоту встречаемости порядка 10^{-4} и ниже. Примерно 400 биграмм (но не одних и тех же для совокупности авторов) не встречаются вовсе.

Расстояния (1.3) от текста до эталона лежат в промежутке от 0,06 до примерно 0,5. Чаще всего расстояние до своего эталона равно 0,15, а до чужого 0,20. Это различие и позволяет определять автора текста с достаточно хорошей точностью. Совместное распределение расстояний до своего и чужого эталонов биграмм показано на Рис. 2. Из него следует, что чаще всего (с вероятностью порядка 0,9) большое расстояние до своего эталона связано с еще большим расстоянием до чужого, что и реализует условие (1.4). Для рассматриваемого корпуса текстов ошибка составила 0,125. Оси расстояний до своего эталона и до чужих эталонов на Рис. 2 разбиты на 20 классовых интервалов с шагом 0,03.

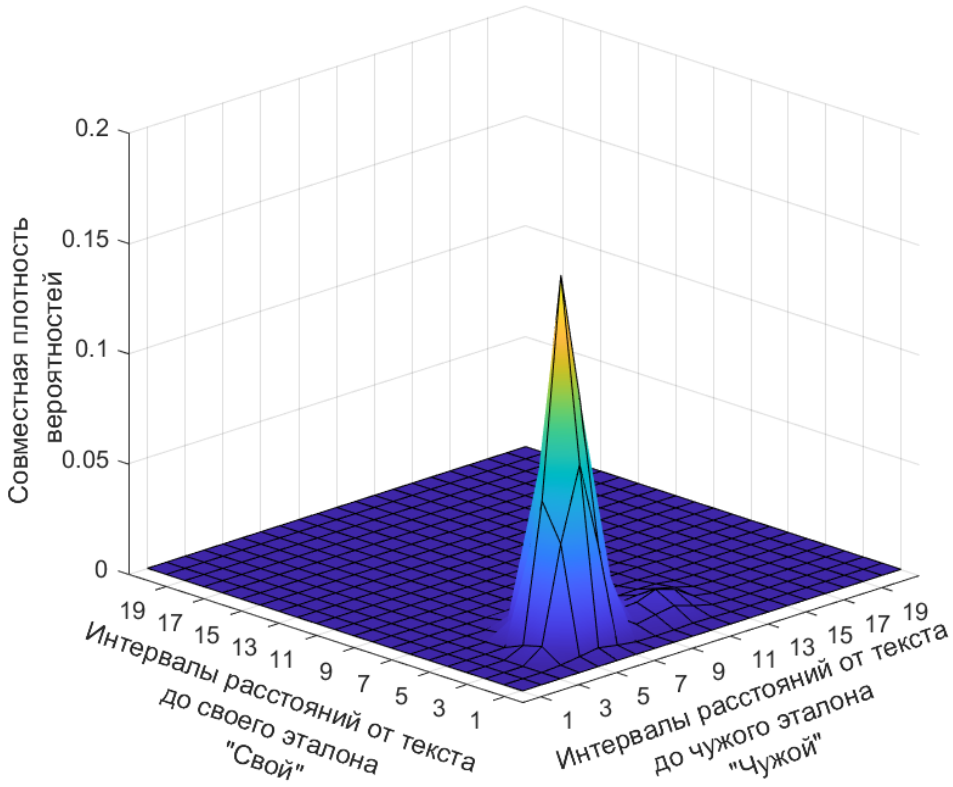

Рис. 2 – Совместная плотность распределения расстояний биграмм между текстами и эталонами «свой-чужой»

Представляет интерес исследование вопроса о том, насколько различные программы предобработки искажают авторское своеобразие текста. То, что они искажают сам текст, естественно. Важно оценить, как меняется при этом возможность идентифицировать автора текста.

2. Анализируемые программы предобработки текстов

Задача автоматической обработки текстов (АОТ) на естественном языке начала развиваться с середины прошлого века. Первые программы были

построены на правилах естественного языка (ЕЯ) и были весьма неточны [10]. С развитием технологий, в частности машинного обучения, стали появляться более совершенные методы работы с ЕЯ.

В настоящий момент АОТ имеет множество приложений: машинный перевод текста, извлечение информации, классификация и кластеризация документов и т.д. Для их реализаций распространена практика предобработки текста. Порядок предобработки текста обычно имеет вид: токенизация, морфологический анализ, синтаксический анализ и семантический анализ [11]. Порядок обработки текста строго определен, поскольку каждый следующий этап использует данные из всех предыдущих этапов. В данной работе анализируются программы предобработки текста на основе токенизации и морфологического анализа.

Токен – это текстовая (буквально – произносимая смысловая) единица, например слово, словосочетание или предложение. Токенизацией является процесс разделения текста на токены. В зависимости от выбора токена используются разные методы токенизации. В данной работе токенизация используется как вспомогательная функция к морфологическом анализу, поэтому далее токеном будет являться слово. Токенизацию можно проводить несколькими методами: простым, то есть разделением слов и предложений по пробелам и знакам препинания, и по правилам. Первый метод обладает преимуществом по скорости работы и предсказуемости результатов. Однако в ряде случаев могут возникать ошибки. Например, если токенизация настроена на разделение слов по пробелам и всем знакам препинания, то такие слова, как «*что-то*» разделятся на «*что*» и «*то*». Если метод не будет разделять слова с знаком «-», то могут возникать ошибки с составными словами, такими как «*малиново-банановый*». На практике обычно используются методы, основанные на правилах. Методы по правилам также разделяют текст по пробелам и знакам препинания, однако в них реализованы дополнительные условия. Это могут быть как список слов-исключений, так и правила пунктуации или все вместе. Главным недостатком таких методов является сложность с предсказуемостью результата. Например, на практике слово «*раз-два-три-четыре-пять*» некоторым методом разделялось на отдельные слова: «*раз*», «*два*», «*три*», «*четыре*», «*пять*», а другим непредсказуемо группировалось на «*раз*», «*два-три*», «*четыре*», «*пять*». Такую ошибку трудно отслеживать, что влечет ошибку при дальнейшем морфологическом анализе.

Морфологическим анализом является определение грамматической характеристики слова (род, число, падеж и т.д.) и нормализация текста [11]. Обычно под нормализацией текста подразумевается приведение слов в тексте к нормальной форме. Нормальная или каноническая форма (например, форма единственного числа, именительного падежа, мужского рода для существительных) также называется леммой. В данной работе нормализацией текста еще будет называться выделение неизменяемой части слова или стемминг. В общем случае стем не является основой слова, например, стемом слова «отомстить» является «отомст», а корнем слова – «мст». Ниже будут подробно описаны некоторые методы нормализации текстов.

Среди множества реализаций методов морфологического анализа текстов были выбраны те, которые являются некоммерческими, популярными и выполнены на языке программирования «Python». Это библиотеки PyMorphy2 [12], PyMyStem3 [13], Stanza (модель syntagrus) [14] и NLTK (модуль Snowball Stemmer) [15]. Их подробное описание представлено ниже.

• PyMorphy2 – библиотека, реализованная в 2012 году Михаилом Коробовым. PyMorphy2 работает только с отдельными словами. Метод морфологического анализа состоит в поиске по словарю «Открытого корпуса». «Открытый корпус» – вручную размеченный корпус на русском языке. Слова в корпусе представлены как совокупность всех их форм и соответствующих им грамматических характеристик. Несловарные слова анализируются с помощью набора правил. Например, в библиотеке хранится список префиксов (таких как «не», «анти», «псевдо»), так что несловарное слово проверяется на наличие такого префикса [12].

• PyMyStem3 – библиотека, созданная компанией Yandex. Данная библиотека проводит токенизацию и морфологический анализ. Метод морфологического анализа основан на машинном обучении по корпусу, представленному в виде деревьев основ и окончаний слов [16]. Обучающим корпусом является часть «Национального корпуса русского языка» (НКРЯ) [17]. НКРЯ содержит около 1,5 млрд слов, часть которых размечена вручную. Исследуемое слово анализируется с конца: сначала в дереве окончаний ищется окончание слова, потом в дереве основ проводится поиск основы слова. Если найденная пара соответствует друг другу, то слово словарное и разбор найден.

• Stanza – мультиязычная библиотека, созданная в Стэнфордском университете. Stanza способна проводить полную предобработку текста, включая токенизацию, морфологический анализ, распознавание именованных сущностей и т.д. Соответствующий метод BiLSTM [18] морфологического анализа основан на машинном обучении. Обучение проводится на корпусе Universal Dependencies, слова в котором представлены в виде дерева. Для русского языке в Universal Dependencies [19] представлены четыре корпуса: UD_Russian-GSD, UD_Russian-PUD, UD_Russian-SynTagRus и UD_Russian-Taiga. Корпус UD_Russian-SynTagRus является рекомендуемым и состоит из вручную размеченной части НКРЯ.

• NLTK – самая старая из представленных библиотек. Она была реализована в 2001 году в Пенсильванском университете. Большая часть модулей библиотеки реализована для английского языка. На русском языке представлен модуль токенизации и стемминга. Самым распространенным методом стемминга является метод Портера [20] (модуль Snowball Stemmer). Он основан на предположении о существовании ограниченного набора словообразующих суффиксов и окончаний. Фактически стемминг не выделяет неизменяемую часть слова, а удаляет суффиксы и окончания из слова. Удаление происходит

по некоторым правилам, целью которых является сохранить основу слова, содержащую минимум одну гласную (для русского языка), и отсечь суффиксы и окончания наибольшей длины.

Пример нормализации текста приведенными выше библиотеками представлен в Таблице 1. Из таблицы видно, что в целом библиотеки нормализуют текст одинаково, но есть некоторые различия. Самое большое отличие отвечает тексту, предобработанному в библиотеке NLTK. Ниже проведен сравнительный анализ точности работы данных библиотек.

Таблица 1. Примеры нормализованного текста

Исходный текст	**PyMorphy2**	**PyMyStem3**	**Stanza**	**NLTK**
Как ныне сбирается вещий Олег	Как ныне сбираться вещий Олег	Как ныне сбираться вещий Олег	Как ныне сбирается вещий Олег	Как нын сбира вещ Олег
Отмстить неразумным хозарам,	Отмстить неразумный хозарам,	Отмщать неразумный хозара,	Отмстить неразумный хозар,	Отмст неразумн хозар,
Их селы и нивы за буйный набег	Они сел и нива за буйный набег	Их сел и нива за буйный набег	Они село и нивы за буйный набег	Их сел и нив за буйн набег
Обрек он мечам и пожарам;	Обречь он меч и пожарам;	Обрекать он меч и пожар;	Обрек он меч и пожарам;	Обрек он меч и пожар;
С дружиной своей, в цареградской броне,	С дружина своей, в цареградский броне,	С дружина свой, в цареградский броня,	С дружина свой, в цареградско броня,	С дружин сво, в цареградск брон,
Князь по полю едет на верном коне.	Князь по поле ехать на верный коне.	Князь по поле ехать на верный конь.	Князь по поле ехать на верном конь.	Княз по пол едет на верн кон.

Сравнение точности работы лемматизаторов является сложной отдельной задачей [21]. Это во многом связано с необходимостью подготовки тестового корпуса большого объема, размеченного или проверенного вручную, тексты которого не пересекаются с текстами из активно применяемых корпусов. Поэтому подобных качественных корпусов на русском языке мало. Кроме того, такие собрания текстов могут использоваться для обучения новых методов лемматизации. Это может привести к ситуации, когда результат расчета точности работы метода недостоверен, поскольку сам метод содержит данные используемого тестового корпуса. Проблема сравнения методов предобработки текстов является главной темой в конференции «Диалог» [21].

Для сравнения точности методов лемматизации был взят тестовый корпус текстов из конференции «Диалог 2020» [22], состоящий из шести типов данных: текстов 17 века, статей из «Википедии», двух тестовых корпусов прошлой конференции «Диалог 2017», части корпуса SynTagRus и части корпуса поэтических текстов Taiga [23]. Данные в тестовом корпусе представлены в виде таблицы слов и соответствующих им лемм и грамматических характеристик. Точность работы рассматриваемого лемматизатора рассчитывается как частное от количества совпадающих букв в леммах, полученных из тестового корпуса и метода морфологического анализа, к общему количеству букв в леммах тестового корпуса. Результат сравнения представлен в Таблице 2. Библиотека NLTK не участвовала в сравнении, поскольку стем и лемма трудно сравнимы. Однако в [11] указано, что точность «Snowball Stemmer» составляет около 80%.

Таблица 2. Точность методов лемматизации для различных корпусов

Название корпуса	PyMorphy2	PyMyStem3	Stanza
GramEval2020-17cent-dev	**0.7465**	0.716	0.7460
GramEval2020-GSD-wiki-dev	**0.940**	0.931	0.923
GramEval2020-RuEval2017-Lenta-news-dev	0.917	0.893	**0.975**
GramEval2020-RuEval2017-social-dev	0.906	0.928	**0.944**
GramEval2020-SynTagRus-dev	0.886	0.928	**0.969**
GramEval2020-Taiga-poetry-dev	0.914	**0.951**	0.912
Общий результат	0.889	0.893	**0.916**

Из Таблицы 2 видно, что данные библиотеки хуже всего работают с текстами 17 века. Такой результат является предсказуемым, поскольку методы морфологического анализа настроены на современный язык. Для четырех корпусов наблюдается сильный отрыв значения точности одной библиотеки относительно остальных. Это можно объяснить тем, что, как сказано выше, лидирующая по оценке точности библиотека, возможно, содержит данные из рассматриваемого тестового корпуса. Несмотря на это, сравнительный анализ библиотек показывает, что точность лемматизации (если не рассматривать корпус текстов 17 века) в среднем выше 90%. Данная оценка хорошо применима к используемому в задаче корпусу текстов, так как он написан на современном языке. Результат влияния предобработки текста на точность работы метода биграмм описан в следующем разделе.

3. Результаты численных экспериментов распознавания автора

Будем называть используемый в данной работе корпус [9] или его тексты оригинальными. Этот корпус состоит из 1783 текстов на русском языке, написанных 100 авторами. Авторы отбирались по единственному принципу –

чтобы у каждого из них было не менее 6 произведений объемом не менее 30 тыс. знаков. Название корпуса, который был предобработан определенной рассмотренной выше библиотекой нормализации слов, будет соответствовать названию библиотеки. В целом предобработанные корпусы будут называться нормализованными. Итак, все слова в текстах оригинального корпуса были нормализованы. Для понимания объема модификации в корпусе была рассчитана суммарная доля изменения в нормализованных словах относительно оригинальных. Она составила: PyMorphy2 – 0,19, PyMyStem3 – 0,25, Stanza – 0,20, NLTK – 0,20. Далее вычислялись ЭРЧ текстов и автора.

На рис. 3-6 представлены ЭРЧ эталонных биграмм авторов нормализованного корпуса. Из рисунков видно, что при лемматизации частота некоторых буквосочетаний сильно выше остальных. Это буквосочетание «ть» с частотой около 0,037, «ат» – 0,02, «то», «на» и «ст» – 0,015. Остальные значения частот буквосочетаний меньше 10^{-3}. При стеме наблюдаются слабые изменения ЭРЧ эталонных биграмм авторов относительно ЭРЧ оригинальных текстов. По порядку убывания наибольшими значениями частот являются: «ст», «то» и «по» около 0.015, «не», «на», «он», «ен», «от», «ра», «ро» – 0.01. Из приведенных выше данных видно, что нормализация текстов сильно меняет распределение эмпирических частот буквосочетаний.

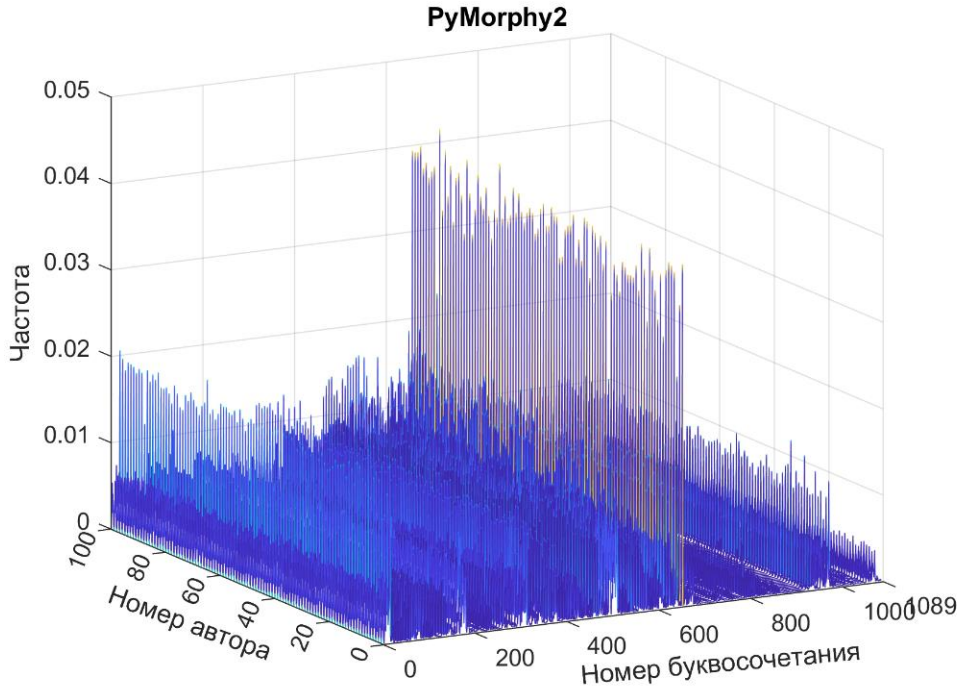

Рис. 3 – ЭРЧ эталонных биграмм авторов нормализованных текстов библиотекой PyMorphy2

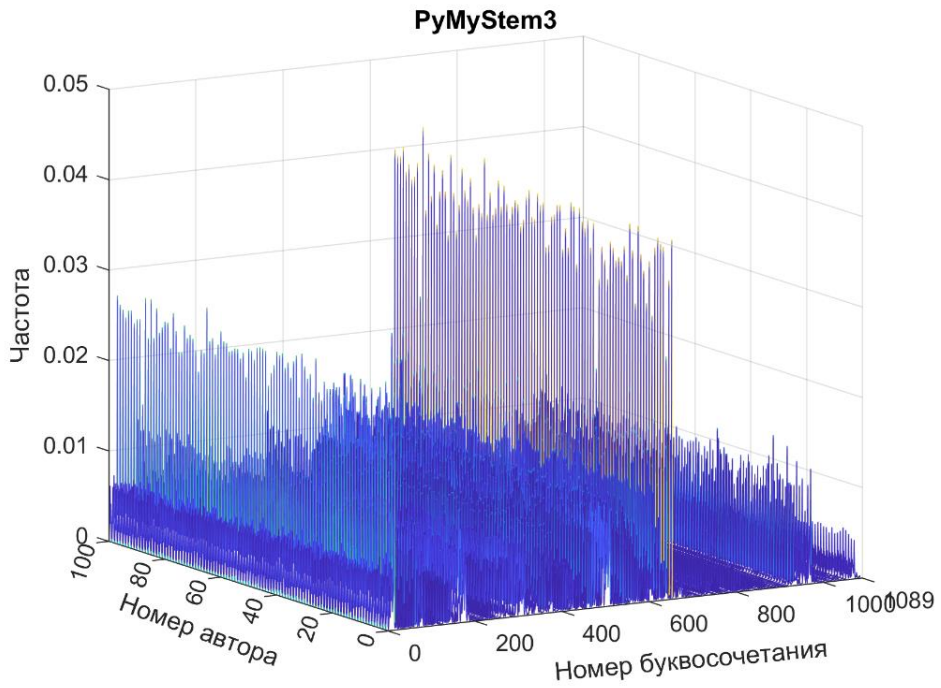

Рис. 4 – ЭРЧ эталонных биграмм авторов нормализованных текстов библиотекой PyMyStem3

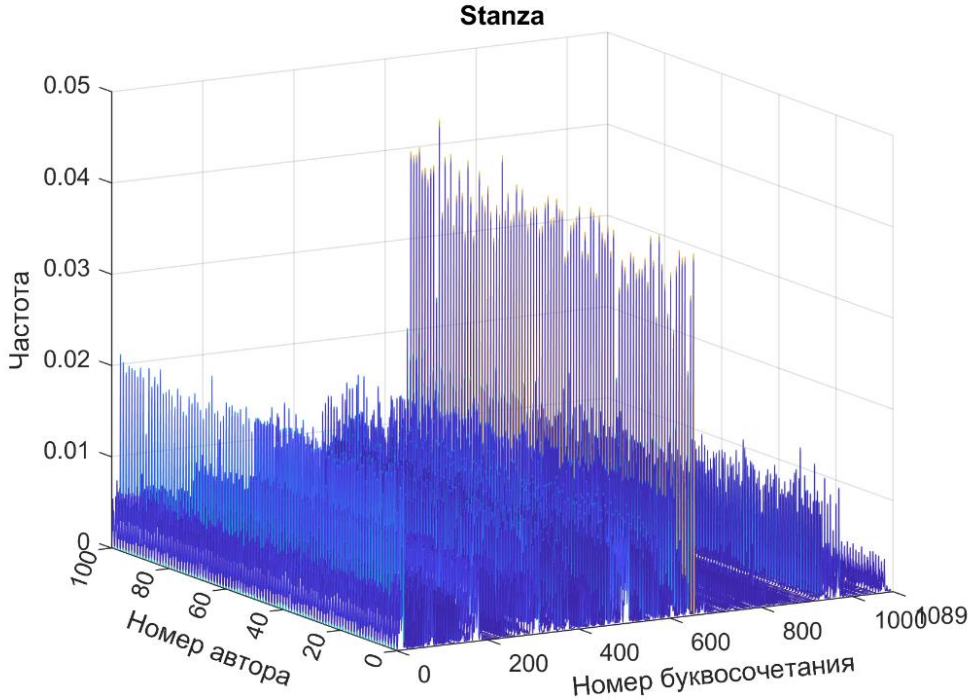

Рис. 5 – ЭРЧ эталонных биграмм авторов нормализованных текстов библиотекой Stanza

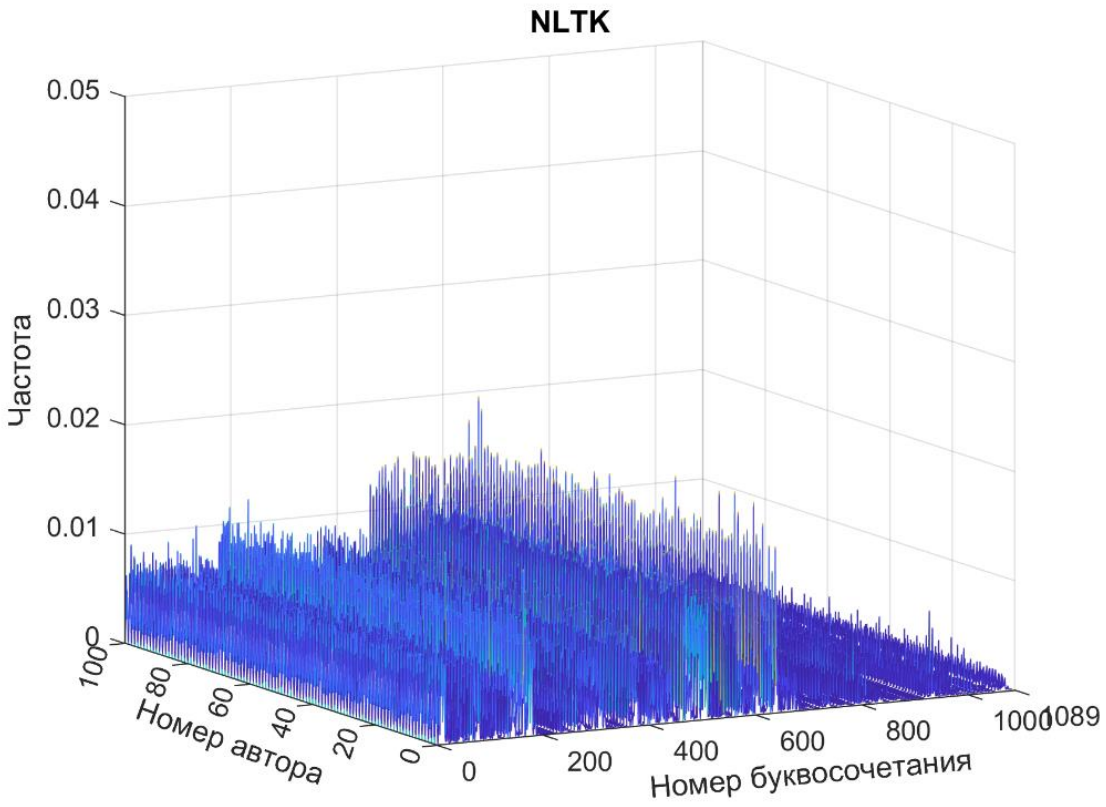

Рис. 6 – ЭРЧ эталонных биграмм авторов нормализованных текстов библиотекой NLTK

Рассмотрим влияние предобработки слов в текстах на точность идентификации автора текста методом биграмм.

Ошибка метода биграмм на оригинальном корпусе составляет 0,125. Соответствующие ошибки на корпусе нормализованных текстов представлены в Таблице 3.

Интересно также сравнить изменение величины ошибки по сравнению с ошибкой распознавания автора оригинального текста с долей измененного текста в результате обработки. Доля измененного текста подсчитывалась в среднем по корпусу текстов следующим образом. Пусть $f_{i,0}(j)$ – эмпирическая частота биграммы j оригинального i-го текста, а $f_{i,k}(j)$ – частота биграммы того же текста в результате его обработки программой типа «k». Тогда долей $\Delta_{i,k}$ измененного текста будем называть разность между распределениями частот оригинального и обработанного текстов в норме L1:

$$\Delta_{i,k} = \sum_j \left| f_{i,k}(j) - f_{i,0}(j) \right|. \quad (5)$$

По корпусу из N текстов доля измененного текста в результате его обработки программой типа «k» определяется как среднее значение долей отдельных текстов:

$$\overline{\Delta}_k = \frac{1}{N} \sum_{i=1}^{N} \Delta_{i,k}. \quad (6)$$

Именно эти величины (6) представлены в таблице 3.

Пусть ε_0 есть ошибка распознавания автора по корпусу оригинальных текстов, а ε_k – ошибка распознавания после предобработки. Чувствительность χ_k ошибки распознавания автора к предобработке текста определяется как относительное изменение ошибки по сравнению с оригинальным текстом, деленное на долю измененного текста:

$$\chi_k = \frac{1}{\bar{\Delta}_k}\left(\frac{\varepsilon_k}{\varepsilon_0} - 1\right). \quad (7)$$

Таблица 3. Ошибка метода биграмм нормализованных текстов

Название корпуса	Точность метода биграмм	Доля измененного текста	Чувствительность ошибки
Исходный текст	0,125	0	-
NLTK	0,124	0,333	-0,024
PyMorphy2	0,137	0,315	0,30
PyMyStem3	0,141	0,318	0,40
Stanza	0,128	0,304	0,078

Из Таблицы 3 видно, что, хотя сама ошибка распознавания меняется незначительно по сравнению с ошибкой для оригинального корпуса, чувствительность ошибки для некоторых программ оказывается очень большой. Тем не менее большой интерес вызывает то, что при существенных изменениях распределений эмпирических частот в результате предобработки ошибка распознавания меняется слабо. Этот факт не является очевидным и свидетельствует о том, что авторы текстов после предобработки идентифицируются практически на том же уровне ошибки, что и в случае исходных оригинальных текстов. Это означает, что авторская конструкция текста в прозе нечувствительна к окончаниям и к использованию служебных слов.

Интересно также, что ошибка метода биграмм на корпусе, нормализованном методом NLTK, уменьшилась. В целом можно сказать, что авторское своеобразие текста практически сохраняется при использовании программ предобработки NLTK и Stanza.

Полученные значения более подробно раскрываются в Таблице 4. В ней результат идентификации текста оригинального корпуса сравнивался с результатом того же текста для нормализованного корпуса. Для наглядности приводятся значения количества текстов.

Таблица 4. Сравнение количества ошибочно идентифицированных текстов оригинального и нормализованного корпусов

Корпусы текстов, сравниваемых с оригинальным	Тексты одинаково ошибочно распознаны на обоих корпусах	По-разному распознанные авторы текстов		
		Количество текстов ошибочно распознано в оригинальном корпусе, но правильно распознано в нормализованном корпусе	Количество текстов правильно распознано в оригинальном корпусе, но ошибочно распознано в нормализованном корпусе	Всего по-разному распознанных текстов
NLTK	180	43	42	85
PyMorphy2	187	36	58	94
PyMyStem3	188	35	64	99
Stanza	184	39	56	95

Из последней таблицы следует, что среди ошибочно идентифицируемых исходных текстов основная их часть по-прежнему распознается неправильно. В целом количество текстов, которые после предобработки стали распознаваться правильно, меньше, чем количество текстов, идентификация которых стала ошибочной. Однако разница несущественная.

В заключение рассмотрим, как изменится точность идентификации автора, если использовать ЭРЧ биграмм текста и эталонных биграмм автора из разных корпусов с одинаковым набором текстов и авторов. Например, ЭРЧ эталонных биграмм автора рассчитывается из оригинального корпуса, ЭРЧ биграмм текста определяется из данных корпуса PyMorphy2.

В Таблице 5 представлены значения ошибок работы метода при таком построении данных. Из нее видно, что при использовании данных из корпуса NLTK как для расчета авторского паттерна, так и для ЭРЧ биграмм текста, ошибка метода биграмм наибольшая. Однако ошибка метода биграмм для корпуса NLTK в Таблице 3 наименьшая. Следовательно, метод биграмм сильно чувствителен к добавлению новых данных в ЭРЧ и устойчив (или возможно улучшается) к их удалению из ЭРЧ.

Таблица 5. Ошибка метода биграмм при использовании ЭРЧ биграмм текстов и авторского паттерна из разных корпусов.

	ЭРЧ эталонных биграмм автора рассчитываются из корпуса					
ЭРЧ биграмм текста рассчитываются из корпуса		Оригиналь-ный	NLTK	PYMORPHY2	PYMYSTEM3	STANZA
	Оригинальный	-	0,453	0,356	0,419	0,323
	NLTK	0,539	-	0,793	0,780	0,758
	PYMORPHY2	0,281	0,437	-	0,163	0,150
	PYMYSTEM3	0,308	0,716	0,171	-	0,166
	STANZA	0,261	0,432	0,144	0,167	-

Заключение

В данной работе были рассмотрены методы предобработки текстов и их влияние на идентификацию автора текста методом биграмм. Предобработкой в данной работе является нормализация слов, выраженная в виде лемматизации и стемминга. Среди реализаций методов нормализации слов были выбраны PyMorphy2, PyMyStem3, Stanza (модель syntagrus) – осуществляющие лемматизацию, и NLTK (модуль Snowball Stemmer) – выделяющий стем слова.

Было выяснено, что при нормализации было модифицировано около 20% букв в рассматриваемом корпусе, что влечет существенное изменение эмпирических распределений частот биграмм авторов. Наиболее частое буквосочетание «то» в непредобработанных текстах становится третьим по частоте после нормализации. Самыми частотными буквосочетаниями становятся «ть» при лемматизации и «ст» при стеме. Кроме того, для таких буквосочетаний меняется порядок значений частот.

Метод биграмм оказался устойчивым к изменениям слов в текстах. При выделении леммы слова ошибка метода биграмм увеличилась незначительно. В случае корпуса, окончания слов которого были удалены, ошибка метода не изменилась. Как было показано выше, устойчивость метода проявляется только при одновременной предобработке идентифицируемого текста и корпуса, относительно которого определяется автор. Предобработка текста и корпуса может выполняться разными методами, главное, чтобы тип нормализации был один (стем или лемма).

Литература

1. Рассел С., Норвиг П. Искусственный интеллект. Современный подход. М.: Вильямс. 2007. 1480 с.

2. Маннинг К., Рагхаван П., Шютце Х. Введение в информационный поиск. М.: Вильямс. 2011. 528 с.

3. Hovy E., Lavid Ju. Towards a science of corpus annotation // International Journal of Translation. 2010. V. 22. No 1. P. 1-25.

4. Toldova S.Y., Sokolova E.G. et al. NLP evaluation: Russian morphological parsers // Computational linguistics and intellectual technologies. Proc. of Int. Workshop Dialog-2010. 2010. Moscow. V. 9. No 16. P. 318-326.

5. Орлов Ю.Н., Осминин К.П. Определение жанра и автора литературного произведения статистическими методами // Прикладная информатика. 2010. Т. 26. № 2. С. 95-108.

6. Орлов Ю.Н., Осминин К.П. Методы статистического анализа литературных текстов. М.: Эдиториал УРСС/Книжный дом «ЛИБРОКОМ». 2012. 312 с.

7. Митин Н.А., Орлов Ю.Н. Статистический анализ биграмм специализированных текстов // Компьютерные исследования и моделирование. 2020. Т. 12. № 1. С. 243-254.

8. Батура Т.В. Методы автоматической классификации текстов // Программные продукты и системы. 2017. Т. 30. № 1. С. 85-99.

9. Кислицын А.А., Орлов Ю.Н. Исследование статистик графов ближайших соседей // Препринты ИПМ им. М.В. Келдыша. 2021. № 85. 23 с.

10. Yngve V. H., Charney E. K., Klima E. S., etc. Mechanical translation. // Research Laboratory of Electronics (RLE) at the Massachusetts Institute of Technology (MIT). 1962.

11. Большакова Е.И., Воронцов К.В., и т.д. Автоматическая обработка текстов на естественном языке и анализ данных: учеб. Пособие. М.: Изд-во НИУ ВШЭ. 2017. 269 с.

12. Korobov M.: Morphological Analyzer and Generator for Russian and Ukrainian Languages // Analysis of Images, Social Networks and Texts. 2015. P. 320-332.

13. Segalovich I. A Fast Morphological Algorithm with Unknown Word Guessing Induced by a Dictionary for a Web Search Engine / Conference: Proceedings of the International Conference on Machine Learning; Models, Technologies and Applications. 2003. USA

14. Peng Qi, Yuhao Zhang and et. al. Stanza: A Python Natural Language Processing Toolkit for Many Human Languages // Association for Computational Linguistics (ACL) System Demonstrations. 2020.

15. Snowball, http://snowball.tartarus.org/ (последняя дата обращения 01.11.2022)

16. Зобнин А. И., Носырев Г. В. Морфологический анализатор MyStem 3.0 // Труды Института русского языка им. В. В. Виноградова. 2015. № 6. С. 300-310.

17. Национальный корпус русского языка. https://ruscorpora.ru/ (последняя дата обращения 01.11.2022)

18. Huang Z., Xu W., Yu K. Bidirectional LSTM-CRF models for sequence tagging. arXiv preprint https://arxiv.org/abs/1508.01991 (последняя дата обращения 1.11.2022)

19. De Marneffe M. C. et al. Universal dependencies // Computational linguistics. 2021. Т. 47. №. 2. P. 255-308.

20. Porter M. An algorithm for suffix stripping // Program: electronic library and information systems. 1980. Vol. 14 No. 3. P. 130-137.

21. Ляшевская О.Н., Астафьева И. и др. Оценка методов автоматического анализа текста: морфологические парсеры русского языка // Конференция Диалог 2010. 2010.

22. Lyashevskaya O. N., Shavrina T. O. et al. Grameval 2020 shared task: Russian full morphology and Universal Dependencies parsing // Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference «Dialogue 2020» Moscow. 2020.

23. https://github.com/dialogue-evaluation/GramEval2020 (последняя дата обращения 01.11.2022)