МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра информатики и программирования

АНАЛИЗ И КЛАССИФИКАЦИЯ РАЗНОЧТЕНИЙ И ОПЕЧАТОК В СЛОВАХ НА ОСНОВЕ РАССТОЯНИЯ ДАМЕРАУ-ЛЕВЕНШТЕЙНА

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студентки 4 курса 441 группы
направления 02.03.03 Математическое обеспечение и администрирование информационных систем
факультета компьютерных наук и информационных технологий
Давыдовой Виктории Вячеславовны

Научный руководитель:
д.т.н., профессор кафедры ИиП _____ А.С. Фалькович

подпись, дата

Зав. кафедрой ИиП:
к.ф.-м.н., доцент _____ М.В. Огнёва

подпись, дата

Саратов 2021

ВВЕДЕНИЕ

Актуальность темы. Каждый день огромное число людей сталкивается с проблемой опечаток при составлении документов, написании научных работ и отправке сообщений. Эти опечатки могут быть совершенно разными и допускаться при большом множестве обстоятельств, однако значительная доля всех допускаемых опечаток приходится на имена, фамилии и отчества людей.

Эта проблема является довольно значимой, поскольку для большого множества организаций хранение базы данных с корректными именами пользователей, клиентов, сотрудников и др. является необходимой мерой для успешной работы. С неуклонным ростом объёмов различной информации и масштабированием баз данных эта проблема встаёт наиболее остро, поскольку выявление опечаток и разночтений является одним из наиболее трудоёмких и дорогостоящих процессов обработки информации.

Однако, несмотря на большое множество уже разработанных методов решения этой проблемы, ни один из них не охватывает полностью допускаемые опечатки в написании имён, фамилий и отчеств людей. Именно поэтому актуальность работы связана с тем, что ни один текстовый редактор или даже редактор-человек не может всегда верно выявлять опечатки, допущенные пользователем при написании имён, фамилий и отчеств.

Цель бакалаврской работы – анализ и предварительная классификация разночтений в словах на основе расстояния Левенштейна.

Поставленная цель определила **следующие задачи**:

1. анализ существующих методов и алгоритмов выявления разночтений в словах на основе расстояний Левенштейна и Дамерау-Левенштейна;
2. анализ исходного фактического материала для определения способов его предварительной нормализации;

3. разработка алгоритма для проверки исходного файла, содержащего конкретные данные и вычисленные расстояния Левенштейна и Дамерау-Левенштейна между ними, на возможные допущенные опечатки и группировка выявленных разночтений по типам.

Методологические основы классификации разночтений и опечаток в словах на основе расстояния Дамерау-Левенштейна представлены в работах Бабакова Р.М., Леонова А.Д., Мазова Н.А., Селезнёва К., Владимирова А.

Теоретическая значимость бакалаврской работы заключается в комплексном анализе большинства уже существующих методов и алгоритмов для выявления опечаток в словах и определении их применимости для выявления разночтений в написании фамилий, имён и отчеств реальных людей.

Структура и объём работы. Бакалаврская работа состоит из введения, 4 разделов, заключения, списка использованных источников и 5 приложений. Общий объем работы – 74 страницы, из них 47 страниц – основное содержание, включая 12 рисунков и 2 таблицы, список использованных источников информации – 20 наименований.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Первый раздел «Методы выявления ошибок в тексте» посвящен рассмотрению основных методов автоматизированного выявления орфографических ошибок и опечаток в текстах.

Известны три основные группы методов автоматизированного выявления орфографических ошибок и опечаток в текстах:

1. статистические методы;
2. полиграммные методы;
3. словарные методы.

Статистические методы основаны на подсчёте количества слов в текстах. В ходе применения статистического метода из текста одна за другой выделяются составляющие его словоформы, а их перечень по ходу проверки упорядочивается согласно частоте. По завершении просмотра текста упорядоченный перечень предъявляется человеку для контроля, например, через экран дисплея.

Полиграммные методы основываются на поиске *полиграмм*. *Полиграммой* или *сограммой* называется фиксированное сочетание букв, встречающееся в том или ином языке в различных словах на различных позициях.

Особое внимание стоит уделить *N-граммным методам* обработки текстовой информации. Именно эти методы являются переходным звеном от статистических методов к полиграммным. Одним из *N*-граммных методов является статистический метод выделения информативных элементов текстов документов и их автоматической классификации, который использует в своей работе частотные характеристики различных фрагментов слов в текстах. Таким образом, метод основан на использовании вероятности появления цепочки букв *N*-го порядка (*N*-грамм) в анализируемых текстах. В качестве классификатора для анализируемого множества документов применяется функция правдоподобия, а значимость тех или иных *N*-грамм определяется по критерию хи-квадрат.

Словарные методы основаны на том, все входящие в текст словоформы, с упорядочиванием или без его использования, в своем первоначальном текстовом виде или после проведения над ними морфологического анализа, сравниваются с содержанием заранее составленного машинного словаря. Если словарь такую словоформу допускает, она считается правильной, а иначе предъявляется пользователю для дополнительного контроля и возможного исправления.

Отличительной особенностью работы является тот факт, что анализируемые данные представлены не в виде текста, а в формате базы данных, содержащей фамилии, имена и отчества людей. Обнаружить ошибку или опечатку в этой ситуации непросто. Действительно, написание имён и, в особенности, фамилий конкретных людей могут отличаться от общепринятых. Так, например, сложно определить, по каким правилам образовываются дагестанские и азербайджанские имена, поэтому их написание в различных источниках может отличаться. Также в неверном написании имён немаловажную роль играет и человеческий фактор.

В соответствии с этим все ошибки можно классифицировать следующим образом:

1. Орфографические ошибки: пропуск буквы, замена буквы, перестановка двух рядом стоящих букв, одна лишняя буква (отдельно может рассматриваться случай удвоения буквы), замена буквы русского алфавита буквой латиницы и др.

2. Морфологические (словоизменительный уровень) ошибки: ошибки в окончаниях (флексиях), несоблюдение правил языка в основе имени или фамилии, новые, ранее не известные имена, отсутствие правил написания имён и т.п.

Несмотря на то, что при использовании статистических методов почти никогда невозможно понять причину, по которой алгоритм сделал тот или иной вывод, именно использование статистических методов может гораздо упростить работу по поиску действительных ошибок в именах, поскольку

они дают высокую точность результатов выполнения. Относительно нейронных сетей, которые тоже применяются как один из статистических методов обработки текста, то в том, что касается непосредственной обработки текста, всегда есть более специализированные алгоритмы, которые решают задачу намного лучше, чем сети.

С точным методом понять причину можно гораздо чаще: есть конкретное правило, которое работает в конкретном случае и позволяет сделать такой вывод. Однако это не гарантирует точного определения ошибок в написании имён, поскольку, как уже упоминалось, не для всех имён существуют чёткие правила написания.

N-граммные методы хороши тем, что являются промежуточным звеном между статистическими и полиграммными методами, благодаря чему дают неплохие результаты, однако они гораздо менее эффективны, чем словарные методы, несмотря на свою мобильность для малых ЭВМ.

Полиграммные методы хоть и менее сложные, чем статистические, и имеют высокую степень обнаружения ошибок, тем не менее, в современной практике используются довольно редко и чаще лишь дополняют словарные методы.

Поскольку предметная область довольно ограничена, то применение подходов, основанных на концептуальных фреймах или словарях гораздо более оправдано, чем синтаксически ориентированные подходы, которые применяются для текстов широкой тематики. Однако к недостаткам этих методов следует отнести трудоемкость и длительность неформализованного этапа подготовки списков имён с вытекающими проблемами по их организации и ведению, что довольно сильно усложняет применение словарей в данной работе.

Ввиду того, что при анализе имён нет необходимости «понимать» их смысл и учитывать особенность естественного языка, данную задачу можно решить вовсе без использования популярной сегодня, но трудоемкой и неэффективной в данном случае компьютерной лингвистики.

В соответствии с вышесказанным можно сделать вывод о том, что ни один из перечисленных методов обработки текста не является идеально применимым в исходном виде к задаче поиска опечаток в фамилиях, именах и отчествах людей. Однако наиболее подходящими для решения задачи являются статистические методы, которые можно доработать с помощью написания собственных алгоритмов и определения необходимых метрик и редакционных расстояний, таких как расстояние Левенштейна и расстояние Дамерау-Левенштейна.

Поскольку для данной работы был выбран статистический метод выявления ошибок в тексте, то любая функция проверки наличия опечаток в словах должна основываться на неточном сравнении. *Неточное сравнение строк* – это процесс поиска похожих, но не в точности совпадающих строк. Степень похожести обычно определяется с помощью расстояний. Так, например, для неточного сравнения строк существуют редакционные расстояния Левенштейна и Дамерау-Левенштейна.

Расстояние Левенштейна – минимальное количество операций вставки одного символа, удаления одного символа и замены одного символа на другой, необходимых для превращения одной строки в другую. Расстояние Дамерау-Левенштейна, как и метрика Левенштейна, является мерой для сравнения «похожести» двух строк. Вариация редакционного расстояния Дамерау-Левенштейна вносит в определение расстояния Левенштейна еще одно правило — транспозиция (перестановка) двух соседних букв также учитывается как одна операция, наряду со вставками, удалениями и заменами.

Второй раздел «Предобработка и нормализация исходного материала» посвящен проведению всех необходимых мер по подготовке исходных данных к дальнейшей их обработке разрабатываемым алгоритмом.

Первым шагом предобработки данных после их извлечения является разбиение на пригодные для работы в коде блоки текста, называемые *лексемами.* Разбиение на лексемы проводилось только для данных,

записанных в столбце «FullName» исходного файла формата .xlsx. Этот столбец содержал полные имена людей. Каждое такое полное имя необходимо было разбить отдельно на фамилии, имена и отчества по пробелам, поскольку другие символы, пригодные для разбиения данных на лексемы отсутствуют.

Также все данные были приведены к нижнему регистру, а туркменские и азербайджанские имена потребовали приведения написания слов «оглы» и «кызы» к слитному с отчествами написанию.

Поскольку данные содержат некоторое число различных символов, цифр, лишних пробелов, их необходимо было удалить, а правильно поставленные дефисы и апострофы оставить.

Данные, записанные в столбце «BirthDate», также было необходимо привести к единому формату для того, чтобы критерий даты рождения можно было использовать при сравнении данных.

Столбец «Sex» также был использован как критерий сравнения данных, однако удобнее и менее ресурсозатратно использовать в качестве обозначения пола цифры, а не строки. Так, строки «Женский» и «Мужской» были заменены на значения 0 и 1 соответственно.

Третий раздел «Разработка алгоритма проверки исходного файла и выявления опечаток» посвящен выбору используемых инструментов и технологий для разработки алгоритма, выделению отдельных групп разночтений и построению алгоритма выявления опечаток и определению их по группам.

Алгоритм разработан на языке программирования Python 3 с использованием Anaconda, а также Google Colab. Python 3 – это мощный инструмент для создания программ, с его помощью можно решать задачи различных типов, также стандартная библиотека включает большой объём полезных функций. Anaconda – это дистрибутив Python и R. Он предоставляет все необходимое для решения задач по анализу и обработке данных (с применимостью к Python). Также Python позволяет работать с

базами данных. Исходные данные были перенесены в базу данных SQLite. Все команды и запросы к базе данных были написаны с использованием языка SQL. При реализации алгоритма анализа опечаток использовались следующие библиотеки: pandas – для анализа и обработки данных, записанных в файл Microsoft Office Excel с расширением .xlsx; sqlite3 – для ускорения работы с данными, записанными в таблицы базы данных.

Далее описан общий алгоритм работы программы. На вход программе поступают данные из файла в формате .xlsx или .xls. В первом столбце таблицы записаны идентификационные номера, во втором столбце записаны фамилии имена и отчества людей, в третьем столбце указаны даты рождения, а в четвертом – пол. Далее для каждой строки таблицы выполняются предобработка и нормализация записей так, как указано в главе 2. Далее для каждой записи необходимо найти её «схожие» написания, то есть имена, написанные с ошибкой, но являющиеся просто другим написанием того же самого имени исходя из аналогичных пола и даты рождения, а также расстояния Дамерау-Левенштейна равного 1. Совокупность этих трёх признаков с большой вероятностью указывает на то, что множество различных, но «схожих» написаний имени принадлежит одному человеку. Для того, чтобы определить такие множества «схожих» имён, из таблицы базы данных выбираются только те пары имён, у которых все три критерия совпадают. Предварительно создаётся база данных SQLite, в ней создаётся таблица, строки которой содержат исходные данные из таблицы формата .xlsx/.xls. Затем для этой базы данных создаётся хранимая процедура, содержащая запрос к базе данных, который выбирает необходимые данные из таблицы: пары имён, у которых совпадают пол, дата рождения, а расстояние Дамерау-Левенштейна между их написаниями равно 1. Все выбранные данные из базы данных записываются в новый фрейм, с которым и осуществляется последующая работа.

Затем с этим фреймом работает разработанный алгоритм, который формирует два дополнительных столбца фрейма, в первый из них

записываются все возможные типы допущенных опечаток, во второй дополнительный столбец записываются буквы, в которых допущены опечатки. Далее этот фрейм можно снова записать в исходный или любой другой формат хранения данных.

В четвёртом разделе «Анализ результатов работы» приведены статистика групп и типов выявленных опечаток и точность работы алгоритма.

По результатам анализа полученной статистики можно сделать вывод о том, что при написании имён люди примерно с одинаковой частотой делают опечатки, связанные как с удалением или добавлением лишней буквы, так и с заменой одной буквы на другую. Стоит уделить особое внимание опечаткам, связанным с заменой гласных «о» и «а» друг на друга, заменой фонетически схожих букв, а также пропускам и добавлениям удвоенных букв, поскольку именно эти опечатки допускаются чаще всего.

Точность работы алгоритма достигает 98,2%. Данный результат говорит о достаточно высоком качестве определения опечаток. Он получился благодаря хорошей предобработке исходных данных и их нормализации.

Однако алгоритм не показал стопроцентной точности своей работы. Эту проблему можно решить путём более детального анализа правил и различных паттернов написания имён разных национальностей на русском языке. Это поможет выявить новые возможные типы опечаток и охватить гораздо большее множество различных написаний имён. А также стоит уделить большее внимание нормализации исходных данных, которые не попадают в написания имён, состоящих из стандартных символов печати. Это однозначно способно повысить точность работы алгоритма.

ЗАКЛЮЧЕНИЕ

В ходе выпускной квалификационной работы были изучены и проанализированы существующие методы и алгоритмы выявления разночтений в словах. При разработке алгоритма выявления опечаток, допускаемых при написании различных фамилий, имён и отчеств, и определения их по группам было использовано редакционное расстояние Дамерау-Левенштейна. Был выполнен анализ и предварительная классификация разночтений и опечаток в словах на основе этого расстояния. Весь исходный материал был проанализирован для определения способов его предварительной обработки и нормализации. Также в практической части работы были разработаны алгоритм выявления опечаток, допускаемых при написании различных фамилий, имён и отчеств и программа, анализирующая подаваемые на вход файлы с исходными данными.

По итогам работы, согласно полученным данным о качестве выявления опечаток, стало ясно, что применяемые методы нормализации и предобработки имён являлись достаточно эффективными. Разработанный алгоритм тоже показал хорошие результаты работы и высокую точность. Однако стоит более детально изучить правила и паттерны написания различных имён на русском языке и дополнить существующую программу. Это однозначно повысит её применимость.

Также способствовать улучшению классификации может более тщательно продуманная первичная обработка текстовых данных. Данное действие также положительно скажется на точности работы программы.

Основные источники информации:

1. Бабаков, Р.М., Леонов А.Д. Методы автоматизированной коррекции специализированных природно-языковых текстов // Информационные управляющие системы и компьютерный мониторинг. - Донецк: ДонНТУ, 2014. - С. 273 - 276.

2. Мазов Н.А. N-граммные методы обработки текстовой информации. // Библиотеки и ассоциации в меняющемся мире: новые технологии и новые

формы сотрудничества: 2-я междунар. конф. Евпатория, 1995. Т. 1. С. 247-250

3. Селезнев К., Владимиров А. Лингвистика и обработка текстов. // Открытые системы. СУБД. 2013, № 4. С. 46-49.

4. Берд С., Кляйн Э., Лопер Э. Обработка естественного языка с Python. – М.: ДМК Пресс, 2015. — 456 с.

5. Джурафски Д., Мартин Д. Обработка речи и языка. – СПб.: Питер, 2017. – 227 с.