Автоматизация обнаружения и исправления опечаток в названиях географических объектов для системы семантического контроля документов электронной библиотеки

Андреев А.М., Березкин Д.В., Нечкин А.С., Симаков К.В., Шаров Ю.Л.

НПЦ «ИНТЕЛТЕК ПЛЮС»
arka@inteltec.ru

Аннотация

В статье изложен метод обнаружения и исправления опечаток в названиях географических объектов. Приведена классификация орфографических ошибок, подробно описан метод проверки и алгоритм, реализующий его. Выполнена экспериментальная оценка и даны направления по дальнейшему усовершенствованию предложенного подхода.

Введение

В настоящее время, благодаря последним достижениям в области информационных технологий, появилась возможность создания эффективных средств для автоматизированного (и даже автоматического) исправления различного рода ошибок в текстовой информации.

Существуют предметные области, в которых задача хранения безошибочных документов в электронных библиотеках имеет особую актуальность. К таким областям можно отнести хранение документов органов государственной власти, банков, ответственных документов коммерческих организаций. В данной работе рассматриваются ошибки в документах, связанные с наличием опечаток, в частности, опечаток в названиях географических объектов (например, в почтовых адресах), и пути автоматического их исправления.

Следует отметить, что в реальной системе проверки и исправления названий географических объектов метод автоматического исправления опечаток должен работать совместно с другими методами анализа и исправления ошибок. Такую систему можно отнести к классу систем семантического контроля, концепция которых излагается в [1, 2]. В частности, при проверке названий географических объектов в составе почтового адреса следует учесть, что адрес часто поступает на вход системы проверки в виде сплошной строки. Для разбора такой строки требуется программа-сегментатор [3], которая должна разбить адрес на составные части (адресные поля). Чем лучше программа-сегментатор разберет входной адрес, тем быстрее и точнее окажется работа алгоритма исправления опечаток в адресах. Кроме того, на выходе возможно получение нескольких вариантов адресов. Для определения правильного варианта, а также для проверки существования указанных номеров домов после алгоритма исправления опечаток используется алгоритм проверки правильности составления адреса, который и подтвердит (или опровергнет) существование полученного адреса. Это лишь упрощенная схема системы проверки адресов. Данную схему можно усложнять и улучшать. Все зависит от требований, которые предъявляются к системе.

Классификация ошибок

Существует несколько критериев классификации всех ошибок, допускаемых в текстах. Среди них можно выделить два основных. Первый критерий, в соответствии с которым ошибки подразделяются на мотивированные и случайные, связан с подмножествами языковых единиц и правил, которые связаны с языком. Мотивированные ошибки допускаются по незнанию этих правил авторами проверяемых текстовых выражений. То есть эти ошибки могут быть проверены и исправлены с помощью применения соответствующих правил языка. Случайные же ошибки являются внешними по отношению ко всей структуре языка. Они могут быть допущены по неосторожности или из-за механических неисправностей (западание клавиш). Такие ошибки нельзя исправить с помощью языковых правил.

Второй критерий классификации ошибок (мотивированных и случайных) связан с языковыми уровнями, нормы (правила) которых оказываются нарушенными в результате речевых ошибок. В соответствии с этим критерием ошибки можно классифицировать следующим образом:

1) орфографические ошибки: пропуск одной буквы, замена одной буквы, перестановка букв, лишняя буква (отдельно может рассматриваться случай удвоения буквы), замена буквы русского алфавита буквой латиницы и др.;

2) морфологические ошибки: ошибки в окончаниях (флексиях) при склонении и спряжении слов, употребление отсутствующих в языке форм

слов, несоблюдение правил чередования в основе, употребление незнакомых вариантов слов, испытывающих колебания в роде или одушевленности;

3) синтаксические ошибки: ошибки в моделях управления слов-предикатов, пунктуационные ошибки, нарушение нормативного порядка слов (в том числе - в устойчивых словосочетаниях), вставка пробела внутрь слова, пропуск пробела (отдельно могут рассматриваться случаи слитного и раздельного написания частиц «не» и «ни»);

4) лексико-семантические ошибки: употребление слов в ненормативном значении, нарушение лексической сочетаемости, семантические противоречия [4].

Орфографические ошибки в текстах делятся на грамматические ошибки и опечатки. Между ними имеется существенная разница. Грамматические ошибки – это ошибки, совершенные по незнанию языка (они входят в класс мотивированных ошибок), а опечатки – это ошибки, допущенные из-за невнимательности (случайные ошибки). Грамматические ошибки часто затрагивают несколько букв слова, в то время как опечатки обычно затрагивают одну букву [5]. Существует четыре типа однобуквенных опечаток:

- вставка буквы (*Москва – Мосоква*),
- пропуск буквы (*Кострома – Костома*),
- замена буквы (*Самара – Сатара*)
- перестановка двух соседних букв (*Псков – Пксов*).

Ошибочное повторение буквы (*Москва – Моссква*) относится к типу «вставка буквы». Кроме того, стоит выделить наиболее часто встречающийся тип неоднобуквенных опечаток – перестановка букв через одну (обычно согласных через гласную, или наоборот, например, *Магадан – Мадаган*) [6]. Грамматические ошибки, сделанные в одной букве, будут расцениваться как опечатки, так как они отличаются лишь тем, что опечатка, в отличие от грамматической ошибки, не может быть проверена правилами языка.

В настоящее время известны три основных метода автоматизированного обнаружения орфографических ошибок в текстах – статистический [7, 8], полиграммный [9, 10] и словарный [4, 11, 12, 13].

Предложенный в данной работе метод относится к классу словарных, поскольку использует словарь эталонных наименований географических объектов. В первую очередь такой выбор обоснован ограниченностью предметной области и возможностью без ощутимого падения производительности сканировать имеющиеся словари.

Характерной чертой нашей предметной области является то, что названия географических объектов являются именами собственными. Такие имена, как правило, не подчиняются грамматике языка, так что зачастую человеку нужно помнить правильное написание каждого имени. С этим и связана основная масса ошибок. В связи с этим, предложенный в данной работе метод направлен на выявление именно опечаток без учета грамматики языка.

Дополнительно стоит отметить, что ошибки в написании имен также могут быть обусловлены неправильным слуховым восприятием. Для исправления таких ошибок кроме грамматики языка и эталонного словаря необходимо иметь фонетический словарь. Но в нашей конкретной задаче данные о почтовых адресах берутся из паспортных данных, поэтому проблемы с восприятием имен на слух не существует, так что в данной работе мы ограничились проблемой исправления опечаток.

Метод автоматического исправления опечаток в названиях географических объектов

Проблема исправления опечаток в названиях географических объектов в отличие от исправления опечаток в обычных текстах имеет свои особенности, которые позволяют упростить задачу. В этом случае можно отказаться от языково-специфических эвристик и от использования экспериментальных данных. Кроме того, в гораздо меньшей степени требуется морфологический анализ и совсем не требуется синтаксический и семантический анализ. Это связано с тем, что, во-первых, названия географических объектов представлены в едином виде (единственное число, именительный или родительный падеж), практически отсутствует флективность языка и, во-вторых, нет смысла прибегать к определению взаимосвязей исследуемых названий в плане синтаксиса и семантики. Определение взаимосвязей между названиями географических объектов, например, составляющих почтовый адрес, можно рассматривать только с точки зрения определения существования данного адреса. Но это не входит в задачи алгоритма исправления опечаток. Соответственно, для исправления опечаток в названиях географических объектов следует применить словарный подход и достаточно ограничиться лишь рассмотрением целого названия, не разделяя его на отдельные составные части – основу, префиксы и аффиксы. В некоторых случаях все же придется прибегнуть к морфологическому анализу, когда название географического объекта представлено в родительном падеже.

С этого может быть предложен метод автоматического исправления опечаток в названиях географических объектов, который состоит из следующей последовательности шагов.

Шаг 1: Проверка входного значения по эталонному словарю.

Шаг 2: Проверка входного значения по эталонному словарю с помощью использования различных вариаций входного значения.

Шаг 3: Проверка входного значения по дополнительному словарю с опечатками.

Шаг 4: Проверка отдельных лексем составного входного значения и восстановление сокращений.

Под входным значением будет подразумеваться название географического объекта, требующее проверки, без типа данного географического объекта (город, улица и др.).

Рассмотрим перечисленные шаги подробнее. Изначально входное значение проверяется по эталонному словарю (шаг 1). В случае нахождения входного значения в словаре данное значение считается правильным и не требует дальнейшей проверки на опечатки. Основным шагом и по сложности и по временным затратам является шаг 2, в основе которого лежит метод исправления орфографических ошибок с помощью перебора, предложенный в работе [5].

Рассмотрим вариант данного метода для опечатки типа «замена буквы». В процессе поиска опечатки входное значение будет модифицироваться. Получаемые модификации назовем гипотезами входного значения (далее просто гипотезами).

Изначально входное значение подается на вход алгоритма исправления опечаток. Данный алгоритм является основой как шага 2, так и всего метода автоматического исправления опечаток в целом. Для применения данного алгоритма требуется, чтобы эталонный словарь, по которому будет происходить поиск, был упорядочен по алфавиту.

Алгоритм исправления опечаток

1. Проверяем входное значение по упорядоченному эталонному словарю и находим место, где входное значение больше предыдущего и меньше последующего экземпляра словаря. Обозначим эти позиции как Prev и Next. Возможны варианты, когда входное значение находится раньше или позже по алфавиту всего словаря. Тогда в качестве Prev и Next выбирается, соответственно, первое или последнее наименование в словаре.
2. Далее сравниваем слова на позициях Prev и Next с входным значением побуквенно, начиная от начала слов. Определяем позиции, в которых слова Prev и Next не совпадают с входным значением. Пусть это будут позиции P и N. Определяем, какая из этих позиций больше, то есть находится дальше от начала слова. Обозначим большую позицию как D. Так как обнаружилось несовпадение входного значения со словом из словаря, то, следовательно, цепочка букв, которую составляют буквы входного значения с первой по букву, стоящую на позиции D, не присутствует в эталонном словаре в качестве начала какого-либо экземпляра словаря. Значит, данная цепочка является ошибочной, и опечатка допущена именно в ней. Следовательно, никакие исправления на участке входного значения за позицией D не приведут к нужной гипотезе. Поэтому ограничим перебор варьированием букв на позициях не больше D. В качестве первой варьируемой позиции V выбираем эту максимальную позицию D.
3. Начинаем цикл перебора. Последовательно подставляем на варьируемую позицию V буквы (возможно еще цифры и знаки) из алфавита. При каждой подстановке символа на позицию V ищем получившуюся гипотезу входного значения в эталонном словаре. Если поиск завершился удачно, и было найдено такое значение, то данная гипотеза сохраняется в специальном выходном стеке и поиск продолжается, так как возможны и другие правильные варианты.
4. После прохождения всего алфавита для выбранной позиции V (то есть, когда закончился цикл вычислений по пункту 3) уменьшаем варьируемую позицию V на единицу и переходим к пункту 3. При V = 0 конец алгоритма. При этом в выходном стеке будут находиться уже исправленные правильные варианты входного значения.

В описанном алгоритме рассматривался только один тип опечатки – «замена буквы». В общем случае требуется, чтобы алгоритм проверял входное значение и на другие типы однобуквенных опечаток. Покажем, что описанный алгоритм может быть обобщен на любой тип однобуквенных опечаток.

После определения варьируемой позиции V вместо замены буквы можно производить вставку, что сразу дает алгоритм исправления опечаток типа «пропуск буквы». Для нахождения лишней буквы требуется поочередно вычеркивать буквы, начиная с позиции V и двигаясь к началу входного значения. Для нахождения типов опечаток, связанных с перестановкой букв, можно переставлять соседние буквы либо буквы через одну, начиная с позиций, соответственно, V+1 и V+2 и двигаясь к началу входного значения.

Отметим что, для перебора не обязательно использовать именно буквенный алфавит. Это может быть любой набор символов, пригодный для данного эталонного словаря.

Важно оценить, какое количество операций строковых сравнений будет выполняться при использовании данного алгоритма в случае нахождения пяти вышеперечисленных типов опечаток. Зададим количество символов в используемом алфавите – A. Количество экземпляров правильных наименований в эталонном словаре – S. Количество операций строковых сравнений при поиске одной гипотезы в словаре (I) может быть различным в зависимости от выбранного алгоритма поиска. Для обнаружения типа опечатки «замена буквы» понадобится A*I*N операций строковых сравнений, где N – расстояние

(количество символов) от начала входного значения до максимальной позиции расхождения D. В итоге для всех типов рассматриваемых опечаток будет определено следующее количество операций строковых сравнений:

- замена буквы – A*I*N,
- пропуск буквы – A*I*N,
- лишняя буква – I*N,
- перестановка соседних букв – I*N,
- перестановка букв через одну – I*N,

Общее количество Sum = (2*A+3)*I*N.

При представлении эталонного словаря в виде упорядоченного вектора можно применить поиск с логарифмической сложностью.

Тогда $I = \log_2 S$, а $Sum = (2*A+3)*N* \log_2 S$.

Существует возможность существенно сократить общее количество операций сравнения за счет изменения не только входного значения, но и экземпляров эталонного словаря.

Обоснование выбора способа изменения входных данных при поиске опечаток

Кроме использования гипотез входного значения, существует вариант поиска, когда входное значение неизменно, а модификация происходит в эталонном словаре. Следовательно, надо определить, какой вариант более удобен, быстр и использует меньший объем оперативной памяти. Причем это следует сделать для каждого вида опечатки отдельно.

В представленном алгоритме с определенного момента используется полный перебор алфавита при замене отдельных букв. Иногда этого может не потребоваться. Например, при поиске лишней буквы во входном наименовании необходимо лишь поочередно вычеркивать по одной букве и сравнивать полученные гипотезы с эталонным словарем. Если изменять экземпляры словаря во время непосредственной проверки входного значения на опечатки, то это займет много времени, так как в худшем случае придется модифицировать весь словарь. Соответственно, количество изменений в словаре при проверке одного входного значения, например на «пропуск буквы», составит S*N.

Если нет возможности модифицировать экземпляры эталонного словаря в процессе работы алгоритма, то следует это сделать до начала работы алгоритма, заранее создав дополнительный словарь некоторых типов опечаток. Этот словарь будет хранить измененные экземпляры эталонного словаря с учетом опечаток.

Какие же типы опечаток надо сформировать в дополнительном словаре, а какие проверять непосредственно вовремя сравнения входного значения с эталонным словарем? Для ответа на этот вопрос следует рассмотреть два способа поиска каждого из рассматриваемых типов опечаток. Первый способ применяет изменение входного значения. Второй способ осуществляет изменения экземпляров эталонного словаря.

В таблице 1 представлены основные типы опечаток и способы их обнаружения, а также приведена оценка числа операций строковых сравнений, необходимых при первом и втором варианте для всех типов опечаток.

Таблица 1.

Тип	Количество операций сравнения		Увелич. размера доп. словаря
	При изменении вх. значения	При создании гипотез в доп. словаре	
Замена буквы	$A*N*\log_2 S$	$\log_2(A* N *S)$	$A* N *S$
Пропуск буквы	$A*(N+1)*\log_2 S$	$\log_2(N *S)$	$N *S$
Лишняя буква	$N *\log_2 S$	$\log_2(A*(N+1)*S)$	$A*(N+1)*S$
Перестановка соседних букв	$(N -1)*\log_2 S$	$\log_2((N –1)*S)$	$(N -1)*S$
Перестановка через букву	$(N -2)*\log_2 S$	$\log_2((N –2)*S)$	$(N -2)*S$

Алфавитная упорядоченность словарей позволяет осуществлять поиск в них с логарифмической сложностью.

Так как нет возможности определить величину N (нельзя в точности определить в каком месте будет сделана ошибка), то для сравнения двух способов выявления опечаток возьмем максимальное N, которое соответствует длине слова L. Если же принять, что опечатка может быть допущена в равной степени в любом месте входного значения, то тогда можно заменить L на $L/2$ при подсчете количества операций строковых сравнений для варианта с изменением входного значения. Но данная замена, как будет видно ниже, при конкретных значениях не будет играть большой роли при выборе способа проверки опечатки.

Если создать дополнительный словарь из всех рассматриваемых типов опечаток, то его размер будет $A*(2*L+1)*S+(3*L–3)*S$. При большом значении A (например, при использовании алфавита русского языка $A = 33$) размер словаря в большей степени будет зависеть от включения в него двух типов опечаток: «замена буквы» и «лишняя буква». Если проверять эти опечатки «на лету» (то есть изменять входное значение непосредственно во время работы алгоритма без использования дополнительного словаря), то размер дополнительного словаря существенно сократится $((3*L-3)*S)$.

Многое зависит от конкретных значений. При разных вариациях A, S, L выбор того или иного способа поиска опечаток может меняться.

Для обоснования выбора способа поиска опечаток в случае проверки названия географического объекта приведем конкретные значения величин A, S, L. Для заполнения эталонного словаря правильными названиями была использована база данных КЛАДР (Классификатора адресов России) [14]. При этом количество

экземпляров в эталонном словаре (*S*) составило 161249, со средней длиной экземпляра (*L*) равной 12 буквам. В задействованный алфавит были включены буквы русского алфавита (А – Я, кроме Ё), цифры (0 – 9), а также еще пять символов, встречающихся в названиях (./()пробел). Размер данного алфавита (*A*) составил 48 символов.

В таблице 2 показаны числовые значения параметров для рассматриваемой задачи исправления опечаток в названиях географических объектов.

Таблица 2.

	Количество операций сравнения		Кол-во гипотез в доп. словаре
Тип	При изменении вх. значения	При создании гипотез в доп. словаре	
Замена буквы	~ 9 964,8	~ 26,47	92 879 424
Пропуск буквы	~ 10 795,2	~ 20,88	1 934 988
Лишняя буква	~ 207,6	~ 26,58	100 619 376
Перестановка соседних букв	~ 190,3	~ 20,76	1 773 739
Перестановка через букву	~ 173	~ 20,62	1 612 490

Выбор способа поиска опечатки зависит от доступного объема памяти и от критичности времени выполнения всего процесса поиска. Если создать дополнительный словарь по всем видам опечаток для анализа входных значений, то требуемый объем памяти составит порядка 3,03 Гб. Данный словарь кроме самих гипотез будет включать в себя также и индексы слов в эталонном словаре. Каждой гипотезе будет прикреплен индекс слова из эталонного словаря, по которому можно будет восстановить правильный вид входного значения. Если же составить словарь без проверки на «замену буквы» и «лишнюю букву», то объем памяти требуемый для дополнительного словаря существенно уменьшится, и будет составлять порядка 81,2 Мб с учетом номеров слов из эталонного словаря. Что же касается времени выполнения поиска опечатки, то при использовании способа поиска опечатки с изменением входного значения для всех типов опечаток оно в несколько раз больше времени поиска по дополнительному словарю. Однако здесь все основное время будут занимать опечатки типа «замены буквы» и «пропуск буквы».

Из таблицы 2 видно, что однозначно можно определиться с двумя типами опечаток. Это «пропуск буквы» – создание небольшого дополнительного словаря (~ 29,5 Мб) более предпочтительно, чем увеличение количество операций сравнения в 517 раз. Второй тип опечатки – это «лишняя буква». Здесь объем дополнительного словаря слишком велик (~1,5 Гб), а разница во времени и количестве операций сравнения незначительна.

Что же касается остальных типов опечаток, то здесь все зависит от выбора способа поиска для опечатки типа «замена буквы». Для поиска опечатки типа «замена буквы» требуется в одном случае дополнительный словарь объемом 1,4 Гб, в другом большое количество операций строковых сравнений, превышающее такое количество при поиске в дополнительном словаре в 376 раз.

Так как в реальных условиях существует возможность хранения в оперативной памяти дополнительного словаря небольшого размера (~81Мб), то выбор способа поиска опечаток для разных типов опечаток будет следующим:

Замена буквы и лишняя буква – изменение самого входного слова;

Пропуск буквы и перестановки букв – поиск с помощью дополнительного словаря.

В результате перед началом работы самого алгоритма поиска и исправления опечаток следует создать дополнительный словарь со всевозможными вариантами гипотез экземпляров эталонного словаря, которые образуются при допущении отдельно трех типов рассматриваемых опечаток: «пропуск буквы», «перестановка соседних букв» и «перестановка через букву». Так для экземпляра «Москва» потребуется 6 гипотез с опечаткой типа «пропуск буквы», 5 гипотез с опечаткой типа «перестановка соседних букв» и 4 гипотезы для опечатки типа «перестановка через букву».

Собственно, на шаге 3 как раз и будет происходить проверка входного значения в созданном дополнительном словаре. При нахождении входного значения в дополнительном словаре по индексу соответствия определяется правильный вид значения в эталонном словаре. Следует отметить, что если входное значение было найдено в эталонном словаре (шаг 1), то поиск опечаток не проводится. Если же входное значение не найдено в эталонном словаре, то последующие шаги будут выполняться независимо от того, будет ли найдена на каком-то из них правильная гипотеза для замены или нет. Поиск опечаток будет продолжен для нахождения всех возможных вариантов опечаток.

Проверка составных названий и восстановление сокращений

Еще одним отличием обнаружения и исправления опечаток в тексте и в названиях географических объектов является то, что текст проверяется пословно и разделителями обычно являются пробелы. Названия же географических объектов часто состоят из нескольких слов (например, *Нижний Новгород*, *Проспект Маршала Жукова*). Данные словосочетания по описанному выше алгоритму исправления опечаток при проверке ничем не будут отличаться от однословных названий, и соответственно есть возможность проверить лишь одну опечатку во всем названии географического объекта. Пословная же проверка позволит исправлять опечатки в каждом слове отдельно. Для такой проверки следует создать

еще два словаря: эталонный словарь, хранящий отдельные лексемы составных названий географических объектов, и дополнительный словарь с опечатками в этих лексемах. Кроме самих лексем данные словари также будут содержать номера составных слов в эталонном словаре. Каждой лексеме, как в эталонном, так и в дополнительном словаре будет прикреплен набор индексов составных слов из основного эталонного словаря. По данным индексам есть возможность восстановить различные полные названия географических объектов, в которых присутствует данная лексема. Разделителями лексем в названиях географических объектов могут являться не только пробелы, но и любые другие разделители, при этом некоторые из них также будут являться лексемами. Так, например, название «Ростов-на-Дону» будет разделено на пять лексем «Ростов», «-», «на», «-», «Дону». Исправление опечаток в лексемах идентично проверке целых названий, за исключением того, что при нахождении нужной гипотезы результатом для замены будет являться не данная гипотеза, а восстановленное с помощью индексации, описанной выше, полное название географического объекта.

Кроме этого при написании составного названия географического объекта, особенно в составе почтового адреса, часто используют сокращения вместо полных названий (*Н. Новгород*, *Ген. Ремизова*). Используя лишь алгоритм исправления опечаток, приведенный в работе [5] и описанный выше, такие названия не исправить, так как их можно отнести в разряд названий с несколькими допущенными опечатками, в данном случае с неоднократным пропуском буквы. Для того чтобы исправить данное название, а точнее восстановить его полную форму, следует: во-первых, разбить название на отдельные лексемы, а во-вторых, осуществить поиск по началу слова по словарю, содержащему правильные отдельные лексемы. В качестве данного словаря есть возможность использовать эталонный словарь, созданный для проверки отдельных лексем. При поиске по началу слова будут отобраны все лексемы из эталонного словаря, которые начинаются на данное сокращение.

Как при поиске опечаток в лексеме, так и при восстановлении полной формы сокращенной лексемы может быть найдено несколько вариантов правильных названий географических объектов, в составе которых присутствует искомая лексема. Проверяемое составное название содержит несколько лексем, и для каждой из них будет найдено определенное множество вариантов правильных названий. Пересечение этих множеств образует результирующий набор искомых вариантов для замены проверяемого названия географического объекта. Совместное использование проверки отдельных лексем и восстановление сокращений позволяет исправлять сложные виды ошибок. Например, название «Н. Новгрод» будет успешно заменено на «Нижний Новгород».

И все же при применении проверки отдельных лексем не стоит отказываться от проверки полных названий географических объектов на опечатки. При пропуске/замене разделителя («СанктПетербург»), только проверка по словарям, содержащим полные названия и варианты этих названий с опечатками, даст положительный результат исправления опечатки.

Экспериментальное исследование работы предлагаемого метода

Как уже упоминалось выше, одним из главных ограничений в работе метода нахождения опечаток является время. Следует разделять два типа времени: подготовительное время и основное время. Подготовительное время – это время, затраченное на загрузку основного и дополнительного словаря в память. Как отмечалось ранее, метод автоматического исправления опечаток в названиях географических объектов работает в составе системы, построенной в соответствии с клиент-серверной архитектурой [1, 2].

Обычно в таких системах загрузка служебной информации, в том числе и словарей, осуществляется один раз в момент запуска и инициализации сервера. Подготовительное время зависит не только от параметров самого сервера, но и от вида представления словаря на жестком диске. Наиболее критичным является время непосредственного анализа в процессе работы сервера. Для оценки этого времени был проведен ряд тестов.

При тестировании входные значения для разных тестовых примеров имели различную длину, содержали разные типы опечаток, допущенные в разных местах. В таблице 3 приведены результаты тестирования (тесты проводились на компьютере с процессором Pentium 4 3ГГц и размером ОЗУ 1Гб).

При проведении тестов происходила проверка названий географических объектов в составе почтовых адресов. Следует уточнить, что под количеством проверенных адресов в таблице 3 подразумевается именно количество полных адресов, а не отдельных адресных полей, включающих в себя названия географических объектов.

Для определения того, насколько точным является представленный метод, было отобрано 4120 реальных адресов. В 97 адресах названия географических объектов не были исправлены. Неисправленные названия можно разделить на четыре группы:

1. Отсутствующие в КЛАДР (В/Ч – отсутствуют в КЛАДР);
2. Сложные сокращения (*СПБ*);
3. Раскрытые сокращения (на входе – *Софьи Ковалевской*, требуется – *С. Ковалевской*);

4. Несколько опечаток в одном слове (*Болгаград* вместо *Волгоград*).

На практике доля ошибок данных типов оказывается невелика, поэтому невозможность их исправления не приводит к ощутимому снижению качества системы. Тем не менее, имеются пути исправления этих ошибок.

Для учета ошибок первого типа необходимо совершенствовать эталонный словарь. Ошибки второго и третьего типа можно устранять за счет ведения отдельного словаря исключений и сокращений. Ошибки четвертого типа являются наиболее сложными, в настоящий момент авторам не известны работы, в которых бы предлагалось применимое на практике решение выявления и исправления множественных опечаток. Однако, для конкретных приложений, таких как проверка названий географических объектов, возможно найти решение путем анализа характерных ошибок и модификации метода исправления с учетом их особенностей.

Таблица 3.

Кол-во адресов	Тип и место опечатки	Время исправления опечатки
1	«Замена буквы» в последних символах значений адресных полей	10 875 мкс
50	Различные опечатки (из реальных данных)	От 487 до 2 385 мкс на адрес
100 000	Различные опечатки (из реальных данных)	6 мин 20 сек
945 684	Различные опечатки (из реальных данных)	1 час
22 459 104	Различные опечатки (из реальных данных)	24 часа

Заключение

Предложенный в статье метод исправления опечаток в названиях географических объектов достаточно прост в реализации. Это позволяет использовать его в различных системах и на различных платформах. Он относится к классу словарных методов исправления ошибок, то есть при проверке используется словарь правильных слов.

Однако, в отличие от универсальных словарных методов исправления опечаток в текстах, данный метод обладает определенной спецификой, в частности, используется дополнительный словарь названий географических объектов с допущенными опечатками некоторых типов. Данный словарь создается перед началом проверки названий и позволяет значительно сократить время проверки. Кроме того, существует возможность исправления, как однословных названий, так и названий состоящих из нескольких слов. Многословные названия географических объектов проходят проверку не только как цельные названия, но и осуществляется их пословная проверка. Это позволяет находить более одной опечатки в названии.

Учитывая, что при написании названий географических объектов часто используют сокращение, в методе предусмотрено восстановление сокращений, что позволяет исправлять, по сути, многобуквенные опечатки, к которым можно отнести сокращение слова до одной-двух букв. Алфавит, применяемый в данном методе, кроме букв русского языка содержит цифры и специальные знаки, встречающиеся в названиях географических объектов.

Выполнен теоретический и экспериментальный анализ вычислительной сложности предложенных алгоритмов. Результаты экспериментов подтвердили хорошее быстродействие и высокую точность рассматриваемого метода.

В плане усовершенствования данного метода в дальнейшем стоит добавить возможность определения наиболее вероятных из множества правильных вариантов для замены названия с опечаткой. Это позволит автоматически выбирать правильное название географического объекта из нескольких представленных вариантов на замену.

Описанный метод может использоваться для нахождения опечаток не только в названиях географических объектов, но в других предметных областях, в которых не учитывается флективность языка и синтаксические связи проверяемых слов. Более того, для данного метода неважно на каком языке и с использованием какого алфавита будет записан проверяемый текст. Главное, чтобы был словарь правильных слов или словосочетаний и соответствующий алфавит.

Литература

[1] Андреев, А. М. Особенности проектирования модели и онтологии предметной области для поиска противоречий в правовых электронных библиотеках / А. М. Андреев, Д. В. Березкин, К. В. Симаков // Электронные библиотеки: перспективные методы и технологии, электронные коллекции: Труды Шестой Всероссийской научной конференции RCDL'2004 (г. Пущино, 29 сентября - 1 октября 2004 г.). – С.93 – 102.

[2] Использование технологии Semantic Web в системе поиска несоответствий в текстах документов / А. М. Андреев, Д. В. Березкин, К. В. Симаков и др. // Электронные библиотеки: перспективные методы и технологии, электронные коллекции: Труды Восьмой Всероссийской научной конференции RCDL'2006 (г. Суздаль, 17 - 19 октября 2006 г.).

– Ярославль: Ярославский гос. унив.-т им. П.Г. Демидова, 2006. – С.263 – 269.

[3] Андреев, А. М. Модель извлечения фактов из естественно-языковых текстов и метод ее обучения / А. М. Андреев, Д. В. Березкин, К. В. Симаков // Электронные библиотеки: перспективные методы и технологии, электронные коллекции: Труды Восьмой Всероссийской научной конференции RCDL'2006 (г. Суздаль, 17 - 19 октября 2006 г.). – Ярославль: Ярославский гос. унив.-т им. П.Г. Демидова, 2006. – С.252 – 261.

[4] Мальковский, М. Г. Прикладное программное обеспечение: Системы автоматической обработки текстов / М. Г. Мальковский, Т. Ю. Грацианова, И. Н. Полякова – М.: МГУ, издательский отдел факультета ВМК, 2000. – 52 с.

[5] Гельбух, А. Ф. Исправление орфографических ошибок с помощью перебора, управляемого морфологическим словарем / А. Ф. Гельбух // Научно-техническая информация. – 1993. – Серия 2, № 5. – С. 23–30.

[6] Гниловская, Л. П. Автоматическая коррекция орфографических ошибок / Л. П. Гниловская, Н. Ф. Гниловская // Культура народов Причерноморья. – 2004. – Т. 2, № 48. – С. 171–180

[7] Сулейманов, Д. Ш. Аналитический обзор отечественных и зарубежных работ обработки естественного языка в аспекте прагматически-ориентированного подхода / Д. Ш. Сулейманов // Информационные Технологии и Телерадиокоммуникации [Электронный ресурс] – Казань, 1999. – Режим доступа: http://www.kcn.ru/tat_en/science/ittc/vol000/st.doc

[8] Гельбух, А. Ф. Эффективно реализуемая модель морфологии флективного естественного языка / А. Ф. Гельбух // Научно-техническая информация. – 1992. – Серия 2, № 1. – С. 24–31.

[9] Мазнов, Н. А. N-граммные методы обработки текстовой информации / Н. А. Мазнов // Материалы 2-ой межд. конф. "Крым-95", Евпатория, Украина, 10–18 июня 1995. – М., 1995. – С. 139–142

[10] Холоденко, А.Б. Использование лексических и синтаксических анализаторов в задачах распознавания для естественных языков / А. Б. Холоденко // Интеллектуальные системы. – 1999. – Т. 4, вып. 1-2. – С. 185–193.

[11] Автоматизация процессов обнаружения и исправления ошибок в текстах / Г. Г. Белоногов [и др.] // Научно-техническая информация. – 1991. – Серия 1, № 7–8. – С. 45–47.

[12] Зиновьева, Н. В. Прикладные системы с использованием фонетических знаний / Н. В. Зиновьева, О. Ф. Кривнова // Проблемы фонетики I. – М., 1993. – С. 288–301

[13] Большаков, И. А. Минимизация перебора альтернатив при автоматизированном исправлении искаженных слов / И. А. Большаков // Семиотика и информатика. – 1990. – № 31. – С. 124–149.

[14] Описание классификатора адресов Российской Федерации [Электронный ресурс] – Режим доступа:http://gisnews.icc.ru/svn/DynWidget/trunk/doc/kladr/opisklad.doc

The method for unsupervized detection and correction of misprints in geographical names for the system of semantic checking and validation of documents

This article describes the method of detecting and correcting of misprints in such special data as geographical names. We give our classification of typical errors and misprints. We pay especial attention on discussing the method itself and proposed algorithm. Some experiments were carried out and results are presented.