2017. № 4 ВОПРОСЫ ЯЗЫКОЗНАНИЯ С./Pp. 115—134
Voprosy Jazykoznanija

МЕТОДЫ ОБНАРУЖЕНИЯ И ИСПРАВЛЕНИЯ ОПЕЧАТОК: ИСТОРИЧЕСКИЙ ОБЗОР

© 2017 **Татьяна Олеговна Шаврина**

Национальный исследовательский университет «Высшая школа экономики»,
Москва, 101000, Российская Федерация; rybolos@gmail.com

В данной работе рассматривается развитие систем автоматического обнаружения и исправления опечаток и требования к таким системам на разных исторических этапах развития. Несмотря на то, что с 1960-х гг. качество исправления неуклонно растет, остаются актуальными основные технологические проблемы: обнаружение слов с опечатками и выбор оптимального кандидата на исправление. Детальный анализ контекстных признаков, таких как символьный контекст, морфологические и синтаксические характеристики, для задач, требующих автоматической коррекции орфографии, может быть полезен для улучшения качества работы систем в сфере обеих проблем.

Ключевые слова: автоматическое исправление опечаток, исправление опечаток, исторический обзор, коррекция орфографии, нормализация текста, словарные опечатки

METHODS OF MISSPELLING DETECTION AND CORRECTION: A HISTORICAL OVERVIEW

Tatiana O. Shavrina

National Research University Higher School of Economics, Moscow, 101000, Russian Federation;
rybolos@gmail.com

This paper discusses the history of methods of automatic spelling correction and the requirements faced by systems implementing such methods at different historical stages. Despite the fact that, since 1960s, the quality of correction has been steadily increasing, two basic technological problems remain: detection of a misspelled word and selection of the optimal candidate for correction. A detailed analysis of contextual features (such as symbolic context, morphological and syntactic characteristics) for NLP-applications utilizing automatic spelling correction can be useful for further improvement of performance in both problematic areas.

Keywords: automatic spelling correction, historical overview, real-word errors, Russian spellchecking, spelling correction, text normalization

1. Вводные замечания

1.1. Проблематика дисциплины

Автоматическая коррекция орфографии — одна из самых сложных проблем в области автоматической обработки естественного языка (natural language processing). Универсального решения для нее до сих пор не существует, однако на протяжении всей своей истории коррекция орфографии отражала актуальные задачи прикладной лингвистики и вбирала в себя самые новые вычислительные методы.

Развитие этой дисциплины определяется двумя основными технологическими потребностями: 1) удобством дальнейшей автоматической обработки текста, подвергшегося корректуре (системы по извлечению фактов, оптимизация поисковой выдачи) и 2) удобством

набора для пользователя (мобильная коррекция орфографии, автоматическая проверка орфографии в текстовых редакторах). Так, в 1960-х гг. в инженерной среде возникла потребность в автоматической коррекции текстов, полученных после процедуры оптического распознавания (optical character recognition). Быстро получив подтверждение полезности такой обработки даже самыми простыми способами, исследователи стали применять автоматическую коррекцию орфографии уже для более широкого спектра задач, и к 1970-м гг. работы и по машинному переводу, и по определению авторства, и по исправлению машинного кода содержали в себе соответствующий модуль. К 1980-м гг. к ним добавились и такие задачи, как распознавание написанных от руки текстов, синтез речи. К этому времени первые коммерческие продукты для редактуры текстов (в частности, электронных писем) стали включать в себя полноценные модули проверки и исправления орфографии. За следующие 10 лет с появлением больших корпусов, доступных для обучения, системы по обработке текста на разных уровнях (морфологический и синтаксический парсинг, снятие семантической омонимии, извлечение мнений, фактов) включили в себя модуль исправления опечаток. К началу 2000-х гг. новая технологическая потребность, связанная с набором текста на мобильных устройствах и предсказывающим вводом текста, спровоцировала появление большого числа самостоятельных продуктов по коррекции орфографии. Сегодняшние модели, несомненно, сильно выигрывают по степени универсальности предлагаемых ими решений, так как обучены на значимо большем объеме данных, с более универсальным набором контекстных признаков и отбирают кандидатов на исправление, основываясь на вероятностных моделях, а не на правилах — однако для русского языка качество их работы еще во многом можно улучшить, особенно принимая во внимание растущее число текстов в социальных медиа (блогах, социальных сетях, форумах), далеких от нормативного языка.

Такие особенности этих текстов, как потенциально неограниченный круг создающих их пользователей, отсутствие проверки при наборе и его быстрота, зачастую ведут к следующим проблемам в ходе стандартной цепочки обработки текстов (краулер — токенизатор — морфоанализатор — лемматизатор — синтаксический анализатор — извлечение именованных сущностей и фактов[1]):

1) неправильная расстановка дефисов и тире, а также неверная обработка сокращений приводят к ошибкам на стадии токенизации;
2) буквенные опечатки приводят к ошибкам при автоматическом морфологическом анализе (далее — морфоанализе) и лемматизации текста, а также к ошибкам при синтаксическом анализе, если в результате этих опечаток изменяется окончание слова;
3) неправильное использование заглавных букв препятствует автоматическому извлечению именованных сущностей и фактов.

Все ошибки типов 1—3 подпадают под определение опечатки, принятое в данной работе: **опечатка** — это любая ошибка набора в тексте, связанная с символами или написанием заглавных/строчных букв.

По отношению к опечаткам системы их автоматической обработки делятся на два типа: первые, программы обнаружения опечаток (spell-checkers), лишь обнаруживают опечатки и сообщают о них пользователю (автопроверка орфографии в браузерах или в текстовых редакторах, например «Microsoft Word»), а вторые, программы исправления опечаток (spell-correctors), самостоятельно принимают решение об исправлении найденных ими ошибок (например, автоматическое исправление в поисковых запросах «Яндекса» [Байтин 2008] и «Google» [Norvig 2010]). Хотя системы второго типа являются логическим продолжением и развитием первых, между ними лежит фундаментальное отличие в сфере их применения: так, если программы обнаружения осуществляют интерактивную проверку набираемого текста и ориентированы на непосредственный контакт с пользователем, то системы исправления

[1] Подробнее об этапах автоматической обработки текста см. [Schäfer, Bildhauer 2013; McEnery, Hardie 2011].

осуществляют проверку и исправление опечаток без такого контакта. В рамках работы с текстовыми корпусами и Рунетом нам, конечно же, более интересен второй тип систем.

Автоматическая коррекция орфографии зачастую понимается более широко, и в расширенном понимании не только опечатки и ошибки подлежат автоматическому обнаружению и исправлению, но и, например, унифицируются различные типы кавычек и тире, происходит контекстное развертывание сокращений [Han et al. 2012], расстановка пробелов после знаков препинания, ликвидация последствий неправильной постановки дефисов и тех же пробелов и т. д. Такая универсальная коррекция используется при создании корпусов [Popescu, Vo 2014] и даже может встраиваться внутрь цепочки обработки текста: например, в работе [Han et al. 2012] описанный модуль применяется на этапе сразу после токенизации для исправления сленга и искажений типа *yyyeeess*, то есть технологическая цепочка принимает вид «токенизация — коррекция — морфологический анализ». В работе [Шаврина, Сорокин 2015] соответствующая задача названа «расширенной лемматизацией».

Подавляющее большинство моделей коррекции орфографии опираются на словари, содержащие списки всех нормативных словоформ. Вопросы использования словарей будут подробнее разобраны далее.

1.2. Принятая терминология

Все слова по отношению к словарю (здесь и далее под словарем понимается список словоформ языка, по выбору — с морфологическими характеристиками и леммами) делятся на следующие группы:

1) словарные — нормативные слова языка, входящие в словарь. При коррекции орфографии рассматриваются как основные кандидаты на исправление слова;
2) несловарные — слова, не входящие в словарь. Их подмножество составляют слова out-of-vocabulary (OOV) — принятый с начала 60-х гг. термин [Kukich 1992] для нормативного слова[2], по каким-либо причинам не включенного в словарь (сленг, неологизм, окказионализм и т. д.). Так, например, слово *привет* в любом словаре русского языка считается словарным, слова *пирвет* и *превед* будут считаться несловарными (см. [Ушаков] и др.), при этом *превед* можно отнести к словам out-of-vocabulary, так как это слово закрепилось и имеет собственную дистрибуцию, отличную от дистрибуции слова *привет*.

Слова, содержащие опечатки, не всегда оказываются несловарными. Часто (по оценкам [Peterson 1986] — от 2 до 16% всех случаев, по данным [Mitton 1987] — до 40%) опечатка ведет к тому, что графический вид слова изменяется от словарного к другому словарному.

(1) *Падал прошлогодний нег*.

Согласно работе [Kukich 1992], по отношению к словарю опечатки делятся на:

1) non-word errors (несловарные опечатки) — опечатки, приводящие графический вид слова к несловарной форме (*конец — кноец*; *лидер — дидер* и т. п.);
2) real-word errors (словарные опечатки) — опечатки, приводящие графический вид слова к другому словарному слову или другой форме того же слова (*орел — осел*; *спел — слеп* и т. п.).

Опечатки также принято [Damerau 1964; Левенштейн 1965] различать по форме: 1) замена (*конь — еонь*), 2) вставка (*конь — конгь*), 3) удаление (*конь — коь*), 4) перестановка (*конь — кноь*).

Как показывают результаты корпусных исследований [Han et al. 2012; Шаврина, Сорокин 2015], в зависимости от источника текстов разнится и доля, и распределение типов

[2] В данном случае под нормативностью не имеется в виду кодифицированность словоупотребления или его стилистическая оценка, речь идет только о включенности слова в стандартный узус.

опечаток. Особо стоит отметить, что вопрос распределения опечаток по типам текстов прежде не рассматривался подробно — ошибки оптического распознавания символов и любой другой технической природы могут присутствовать даже у самых «надежных» в плане корректуры изданий. Машинные ошибки, в отличие от человеческих, редко бывают распределены случайным образом, а значит, представляют опасность для корпусных лингвистов, доверяющих результатам автоматической разметки материалов таких изданий без модуля коррекции. Наиболее значимые контекстные признаки для универсальной системы коррекции орфографии еще только предстоит выявить и изучить.

2. Ранние попытки автоматической коррекции

Как сообщает С. Осовцев [1999: 6], у московского библиофила В. Лаврова есть раритет: «Первый альманах», отпечатанный в Смоленске в 1930 г. На обложке размашистым почерком написано:

> *Просьба не замечать опечаток и не сердиться. А. Твардовский. 12.IX.30. Смоленск. P. S. Господа читатели! О всех замеченных в данной статье опечатках просим сообщить в редакцию или автору. Вознаграждение гарантируется.*

Со времен первопечатника Иоганна Гутенберга, который, к слову, и сделал первую задокументированную опечатку в истории (в 1460 г. в латинской грамматике «Католикон» вместо слова *quos* было напечатано *qnos*: к ошибке привела случайно перевернутая буква *и* [Шерих 2004]), появление опечаток в печатных изданиях и их последующее исправление в следующих выпусках стало привычным для читателей. Однако с ходом технического прогресса изменились и условия работы издателей и корректоров.

Первые попытки исправлять опечатки в тексте автоматически относятся к началу 1960-х гг. Эта задача решалась на втором поколении компьютеров, однако не то что исправить — найти опечатку оказалось труднее, чем ожидалось.

Решение оказалось тесно связано с качеством распознавания текста. Оптическое распознавание символов развивалось параллельно и не обладало требуемым уровнем качества — чем хуже оказывалось качество распознавания текста, тем ниже было и качество обнаружения ошибки [Hanson et al. 1976]. Ученые попали в ловушку: даже если бы на тот момент качество распознавания символов достигло 99-процентной точности (одна ошибка на 100 символов), то точность распознавания слов при этом не превысила бы 95-процентного потолка [Kukich 1992], если брать в расчет язык типа русского или английского, в которых средняя длина слова равна пяти буквам.

Постепенно появлялись первые интерфейсы для рукописного ввода текста, а также устройства для голосового ввода команд. Распознавание текста стало каждодневной реальностью, однако оно оставалось несамостоятельным без систем автоматической коррекции [Ibid.]. Программная проверка орфографии стала ключевой проблемой в большинстве актуальных задач, связанных с оцифрованными текстами, таких как машинный перевод, определение авторства, автоматическое исправление кода и др., важной частью архитектуры всевозможных программ для изучения языков, приложений с голосовым вводом и выводом. Будучи в положении ключевой дисциплины, проверка орфографии разделилась на два основных направления работы: обнаружение опечаток (error detection) и их автоматическое исправление (error correction). Исследователи фокусировались на трех основных группах проблем: 1) обнаружение несловарных опечаток (non-word error detection) — первые попытки решения относятся к 1970—1980-м гг.; 2) исправление опечаток в отдельных словах (isolated-word error correction) — с 1960-х гг. и 3) исправление опечаток в зависимости от контекста (context-dependent word correction) — с 1980-х гг.

Основным методом на начальном этапе вплоть до 80-х гг. был n-граммный метод: на некотором словаре или корпусе текстов считались вероятности буквенных n-грамм (обычно

2- или 3-буквенные), и несловарные слова исправлялись на словарные варианты согласно этим вероятностям и наименьшему расстоянию Левенштейна.

Со временем, применяя помимо n-граммных вероятностей позиционные правила (фонетические ограничения на начало слова, конец слова, сочетания согласных и гласных, характерные для каждого конкретного языка), исследователи смогли повысить точность исправления английских слов до 98% (на выборке из 8000 шестибуквенных слов, не связанных синтаксически; каждое из которых содержало одну ошибку [Hanson et al. 1976]).

В итоге к середине 1970-х гг. на рынке появились первые коммерческие системы, способные достаточно хорошо исправлять простые опечатки. Основной проблемой остались словарные опечатки и ошибки, неисправимые без учета контекста: примеры вида англ. *see you in five minuets* (букв. ‘увидимся через пять менуэтов’) регулярно просачивались в тексты. Многие редакторы считали, что использование программ для коррекции орфографии только увеличивает количество сбивающих с толку ошибок, ср. также цитату из письма Сергея Довлатова (как известно, в своих отпечатанных текстах писатель из принципа исправлял опечатки чернилами и приносил обратно издателю): «Гранки получил, спасибо. Заметил несколько мелких опечаток. Почти все они перешли из книжки “Компромисс”, которая в свою очередь — репринт с Гришиного альманаха. И какие-то две-три добавили вы сами. Но мама и Лена еще не читали. Они, я уверен, еще что-нибудь найдут» (из письма к Игорю Ефимову, 17 августа 1984 г.).

Так или иначе, даже освободившись от проблемы оцифровки текста, автоматическая коррекция не приобрела статуса «панацеи корректора».

3. От системы правил к машинному обучению

Исследование n-граммных буквенных моделей продолжалось и после 1970-х гг., но привело исследователей в тупик: заманчивая идея, что можно построить биграммную или триграммную модель языка, добавить ограничения на структуру слога, начало и конец слова и отказаться от использования словаря, привела к тому, что вместо обнаружения ошибок такие алгоритмы плодили новые опечатки. С тех пор большинство методов автоматической коррекции орфографии основываются на использовании словарей. Точку в вопросе эффективности символьных n-грамм для обнаружения опечаток поставила работа [Zamora et al. 1981]: на большом корпусе опечаток была построена отдельная языковая модель и было доказано, что значимой статистической разницы между триграммами нормативных слов и опечаток нет.

Новая методика, однако, принесла и новые сложности — ученые попали в положение, при котором сложности создавала уже не модель, но словарь: оказалось неясным, способна ли простая сверка со словарем решить проблемы обнаружения слов с опечатками (error detection). Дополнительной задачей оказалась оценка требуемого объема лексикона: как правило, большое количество слов на конце частотного списка составляли длинный «хвост», который существенно осложнял выбор правильного кандидата на исправление опечатки. Так, по результатам исследования [Peterson 1986] при объеме словаря больше 350000 слов каждая сотая опечатка совпадает со словарным словом, поэтому не может быть обнаружена простым сравнением со словарем. С увеличением объема словаря количество таких ошибок вырастает до 16%. К 1990-м гг. исследователи стали придерживаться мнения, что для словарь должен содержать только самые частотные лексемы.

В [Kukich 1992] выделено три типа опечаток: 1) типографские (вызванные близостью клавиш друг к другу); 2) фонетические (омофоны); 3) когнитивные (чаще всего в этот тип попадают паронимы). Впрочем, проблемы классификации опечаток отходят на второй план, когда есть несовершенства в системе их поиска и выбора кандидата на исправление. В конце 1980-х гг. доминирующими в рассматриваемом направлении окончательно стали вероятностные методы. В период 1980—1990-х гг. обычно все типы ошибок обрабатывают одинаково, основываясь на одной вероятностной модели; исключением можно назвать работу

[Toutanova, Moore 2002], где строится две модели поиска кандидатов — для типографских и фонетических опечаток.

Во-первых, буквенные n-граммы, показывавшие неплохое качество исправления опечаток (но не их обнаружения), стали зачастую дополняться моделями, учитывающими в том или ином виде ошибки (учитывались вероятности различных перестановок, замен, вставок и удалений, иногда в зависимости от контекста). Во-вторых, лучшие кандидаты на исправление (обычно использовалось расстояние Левенштейна не более 3) начали выбирать, используя скрытые марковские модели [Oshika et al. 1988]. Для обучения и получения вероятностей слов и n-грамм использовалось обычно два разных корпуса: корпус обычных текстов (для обучения n-граммной модели, language model) и корпус слов с опечатками (для настройки модели опечаток, error model).

Полученные в результате модели оказались достаточно плодотворными и стали применяться и для других целей: так, в работе [Oshika et al. 1988] при помощи скрытой марковской модели удалось создать классификатор имен собственных для пяти языков, см. также работу [Hanson et al. 1976]. Проверка орфографии снова стала смежной задачей с оптическим распознаванием образов, существенно повысив его качество.

В 1980-х гг. для исправления опечаток впервые, и весьма успешно, были использованы нейронные сети, обученные с помощью метода обратного распространения ошибки (backward propagation) [Rumelhart et al. 1986]. Их стали применять для ранжирования кандидатов — и в 1986 г. при помощи метода обратного распространения ошибки была успешно обучена трехуровневая сеть. Этот метод дал возможность разработчикам нейронных сетей выбрать набор признаков и их весов, позволяющий определить наиболее правильного кандидата. Обучение производилось на корпусе с ошибками и их исправлениями. Успешное применение нейронных сетей придало импульс их использованию и в других областях прикладной лингвистики, и их стали использовать для распознавания написанных от руки текстов [Burr 1987], для любого вида оптического распознавания символов, а также для синтеза речи [Kukich 1990] и даже автокоррекции адресов на почте [Gersho, Reiter 1990].

К началу 1990-х гг. ученые пришли к следующим важным выводам [Deffner et al. 1990]:

1) нейросети, обученные на корпусе без опечаток, обучались хуже, чем нейросети, обученные на корпусах, содержащих опечатки;
2) при увеличении количества скрытых уровней нейросети качество обучения неизменно растет вплоть до 183 уровней. При большем числе уровней качество падает;
3) качество коррекции орфографии возрастает, если в числе прочих признаков добавить морфологическую и семантическую информацию.

4. Переход к контекстным моделям

К концу 1980-х — началу 1990-х гг. программы коррекции орфографии были попрежнему далеки от идеала. Хотя алгоритмы исправления изолированных слов (isolated-word correction) успешно развивались, ни одна программа не отвечала следующим критериям:

1) большое лексическое покрытие (более 100 000 лексем);
2) независимость качества исправления от длины слова (для коротких слов больше словарных кандидатов);
3) высокая скорость работы и возможность онлайн-исправления;
4) способность успешно выбирать между несколькими кандидатами на исправление.

В статье [Kukich 1992] приводится таблица точности (accuracy) разных методик коррекции орфографии на состояние начала 1990-х гг. — не все из них напрямую сравнимы, однако их все возможно было протестировать на одних и тех же выборках.

Таблица 1

Аккуратность некоторых контекстно-независимых методов исправления слов по работе [Kukich 1992]

Метод	521-словная тестовая выборка	1142-словная тестовая выборка	1872-словная тестовая выборка
Minimum Edit Distanse			
grope	64%	62%	60%
Similarity Key			
Bocast Token Reconstruction	80%	78%	75%
Simple N-gram Vector Distance			
Dot Product	58%	54%	52%
Hamming Distance	69%	68%	67%
Cosine Distance	76%	75%	74%
SVD N-gram Vector Distance			
Cosine Distance	81%	76%	74%
Probabilistic			
Kernighan-Church-Gale Error Probs	—	78%	—
Neural Net			
Back-Propagation Classifier	75%	75%	—

Ни один из методов не дал качества выше 81%. Для тестовой выборки подбирались предложения как с опечатками, сгенерированными при оптическом распознавании, так и допущенными людьми. Стоит отметить, что большинство представленных систем делают качественный рывок (более чем на 10%), если для исправления брать не только самого вероятного кандидата, но множество из трех наиболее вероятных вариантов. В этом случае достижима точность выше 90%. Эти результаты показали необходимость улучшения вероятностной модели отбора кандидата. Также очевидно, что это невозможно без применения контекстных методов.

Другой сложной задачей, существенно продвинувшей вперед развитие исправления опечаток, стало исправление так называемых real-word errors. Рассмотрим подробнее природу подобных ошибок. Согласно [Kukich 1992], часть из них происходит из-за тех же неточностей набора, что и обычные опечатки (например, *from → form*), другая часть определяется фонетическими или когнитивными закономерностями (например, *your → you're*); оставшаяся же относится к морфологическим, синтаксическим и семантическим ошибкам (уже рассмотренные примеры типа *see you in five minuets*, неправильные личные окончания и т. д.). Заметим, что в этой классификации отсутствуют еще два типа: опечатки, вызванные неправильной постановкой пробела или дефиса (*myself → my self*, *nineteen ninety-nine → nineteen-ninety-nine*), а также контекстные ошибки, состоящие в пропуске слова. От того, к какому типу относится большая часть неохваченных опечаток в выборке, зависит и то, какой контекстный метод будет наиболее эффективен.

Для обнаружения и исправления всех указанных типов ошибок может оказаться полезной контекстная информация, однако в разных случаях требуется разный контекст (морфологический, синтаксический, семантический и т. д.). В языках со строгим порядком слов (а вся индустрия коррекции орфографии, безусловно, англоцентрична) наиболее эффективными показали себя n-граммы словоформ (далее под «n-граммами» мы будем подразумевать

только их, если не оговорено иное) и частей речи, однако в случае с русским языком таких сведений явно недостаточно, и нередко добавляют информацию, полученную из синтаксической разметки (см. об этом подробнее в разделе 6).

К началу 90-х гг. качественная контекстная проверка орфографии оставалась недостижимой целью даже для английского языка. Однако, как это часто бывает в прикладной лингвистике, к улучшению привел прорыв в соседней области — в это время для автоматического синтаксического разбора стали использоваться не только системы правил, но и «релаксационный подход» (relaxation-based approach) [Kukich 1992]: многие ранние NLP-системы опирались на строгий набор синтаксических правил и потому часто выходили из строя при их нарушении; в relaxation-based методах при ошибках парсинга проверяется, может ли игнорирование того или иного нарушенного правила привести к успешному разбору. Информация о синтаксических связях и их нарушении оказалась полезной и для исправления опечаток [Deffner et al. 1990].

Так, две системы IBM EPISTLE и CRITIQUE [Heidorn et al. 1982; Richardson, Braden-Harder 1988], построенные для английского языка и использующие при исправлении опечаток гипотетическое синтаксическое дерево, а также аналогичная система для нидерландского [Kempen, Vosse 1990] показали неплохие результаты на достаточно «грязном» тексте (86% точности на обычных наборных текстах и 46% точности на узкопрофильных текстах с профессионализмами). Обе системы были построены на наборе синтаксических правил, предлагая наиболее вероятного кандидата в случае их нарушения, то есть исправляли как несловарные, так и словарные ошибки, однако последние — в пределах грамматики (ошибки согласования, управления и неверное словоизменение). Набор из 300 правил позволял исправлять ошибки даже в тех случаях, когда расстояние Левенштейна между кандидатом и исходным словом было велико.

Система для нидерландского языка использовала фонетические триграммы и их вероятности, а также набор из 500 синтаксических правил, применяемых, однако, не к целым предложениям, а в основном к входящим в них клаузам. Похожая схема обработки (препроцессинг: токенизация, морфологический анализ — синтаксический анализ: поиск ошибок — ресинтез текста на основе правил) использовалась и в некоторых коммерческих системах [Van Berkel, DeSmedt 1988].

Еще более успешным оказался развившийся в то же время статистический подход, основанный на n-граммных языковых моделях (language models). При его применении учитывались вероятности вхождения слова в зависимости от n предыдущих слов. В основном в работах 1990-х гг. использовалась следующая контекстная информация (часто разные источники информации комбинировались):

1) триграммы по словоформам (какое наиболее вероятное продолжение фразы *Шел проливной…*?);
2) триграммы по частям речи (что чаще всего идет после существительного/местоимения в дательном падеже и модального глагола?);
3) степень лексической сочетаемости между словами, стоящими недалеко друг от друга, прежде всего меры сходства между соответствующими им векторами в vector space model (косинусное сходство, коэффициент Отиаи).

Стоит отметить, что данные методы активно применялись не только для исправления орфографии, но и для распознавания текста [Bahl et al. 1989; Jelinek et al. 1991], извлечения фактов [Choueka 1988], генерации текста [Smadja 1991], синтаксического парсинга [Church 1988], машинного перевода [Brown et al. 1990] и снятия семантической неоднозначности [Brown et al. 1990]. Применение статистических методов было бы невозможным без значительного роста вычислительных мощностей, который позволил использовать как большие наборы данных, так и более сложные вычислительные алгоритмы. Так, частоты для статистических моделей таких систем определялись по «большим корпусам» (big corpora) объемом не менее 10 млн слов.

Однако, вследствие закона Ципфа [Zipf 1949], если все слова корпуса упорядочить по убыванию их частоты, то частота n-го слова в таком списке окажется приблизительно обратно пропорциональной его порядковому номеру n (так называемому рангу этого слова). Каким бы большим ни был корпус, подавляющее большинство самых редких слов и тем более n-грамм встретятся в нем всего один раз, то есть по ним нельзя будет получить достоверных частот. В дальнейшем для обхода этой проблемы зачастую использовали униграммные и биграммные модели дополнительно к триграммам, а также использовали сглаживание (чаще всего для n-грамм, не встретившихся по корпусу, вероятность искусственно делают ненулевой, но очень маленькой). Более подробно данная проблематика разрабатывалась в исследованиях по статистическим языковым моделям [Chen, Goodman 1998].

Одной из самых удачных коммерческих систем конца 1980-х гг. можно считать программу CLAWS (Constituent Likelihood Automatic Word-tagging System) [Garside et al. 1987], успешно совмещавшую три подхода: словарный, морфологический и синтаксический. Для вычисления вероятностей как морфологических триграмм, так и синтаксических биграмм, использовался тщательно размеченный вручную корпус объемом в 1 млн словоформ. Каждое словарное слово получало разметку двух уровней — морфологическую на основе словаря и синтаксическую на основе биграмм, а каждое несловарное слово — морфологическую и синтаксическую на основе n-грамм. Если каких-то n-грамм было недостаточно для точного определения морфологической метки, то модели дополняли одна другую, и недостающая метка определялась по имеющимся данным — либо морфологическим, либо синтаксическим. Точность приписывания метки, на котором основывается повторный синтез текста, доходила до 96%.

Данная модель потенциально позволяла исправлять 1) несловарные опечатки, 2) словарные опечатки в неправильных синтаксически контекстах, будь то локальных или глобальных, 3) словарные опечатки в правильных синтаксически контекстах, но неправильных семантически, и достигала точности свыше 90%, однако полнота оставалась низкой (максимум 47%) [Garside et al. 1987].

И хотя статистические модели были еще несовершенны (вспомним известный пример с переводом документации Microsoft для драйвера мыши Windows95, которую переводчик «Poliglossum» с медицинским, коммерческим и юридическим словарем превращал в «Гуртовщика мыши»: *Если вы пользователь Microsoft мыши посетите Microsoft Слугу Паутины, где в особом подвале вы сможете опустить-загрузить самого текущего гуртовщика Microsoft мыши*), их применение стало настоящим прорывом, подняв точность до недостижимых ранее показателей и существенным образом повлияв на развитие науки и индустрии в области исправления опечаток.

5. Актуальное состояние индустрии

С середины 2000-х гг. автоматическое исправление опечаток вступило в новую эру. Во-первых, изменились к лучшему возможности обучения моделей, так как стало доступным использование больших корпусов (одним из первых исследователям стал доступен [BNC], содержащий более 100 млн слов), а во-вторых, продукты по исправлению опечаток стали гораздо более востребованы на массовом рынке: с распространением персональных компьютеров у каждого пользователя появилась как минимум одна мультиязычная программа такого рода — это или модуль коррекции орфографии в «Microsoft Word», или свободно распространяемые «aspell» и «uspell», используемые в UNIX-системах. С недавних пор для русского языка доступен и продукт «Яндекса», разработанный с учетом специфики русского языка — «yspell». Такие программы рассчитаны, прежде всего, на обнаружение опечаток, тогда как автоматическое исправление осуществляется в основном в веб-интерфейсах: обработка поисковых запросов у «Яндекса» [Байтин 2008] и «Google» [Norvig 2010] обязательно включает в себя орфографическую коррекцию.

Ниже приводится таблица 2 с оценкой полноты таких систем на тестовом корпусе в 1000 вхождений по материалам из «Живого Журнала» [Шаврина, Сорокин 2015: 9].

Таблица 2

Полнота исправления опечаток на корпусном материале

Программа	yspell	aspell	Google	MS Word 2007	GICR lemming	Яндекс
Полнота на тестовом корпусе опечаток	0,862	0,694	0,765	0,709	0,748	0,829

Как видно из таблицы 2, для русского языка все эти системы не предоставляют достаточной полноты (хотя бы 90%).

К концу 2000-х гг. немалые усилия в индустрии проверки и коррекции текста были направлены на развитие мобильной отрасли: популярный сегодня способ коммуникации при помощи чатов и SMS оказался весьма неудобным без автоматической корректуры — набор текста на маленькой клавиатуре приводит к тому, что даже самые аккуратные и неторопливые пользователи должны проявлять особое внимание для точного выражения своей мысли. Самыми известными в этой сфере системами стали T9 (от англ. **T**ext on **9** keys), «ITap» и «Swype».

Первенство в сфере мобильного набора текста принадлежит T9 компании «Tegic Communications», выпустившей первый телефон с функцией предугадывания и исправления набора уже в 1999 г., однако программе-первопроходцу пришлось пройти через этапы совершенствования, чтобы достичь результатов, воспринимаемых сегодня как данность. Так, первые версии T9 не могли приспособиться под нужды пользователя, а также не умели исправлять склейки слов из-за пропущенного пробела. Предугадывание слов происходило не лучшим образом: предлагались слова-кандидаты только той длины, что уже была набрана. Ошибки, изменявшие длину целевого слова, были неисправимы.

Программы «Swype» от «Nuance» и «ITap» от «Motorola» более успешно решали проблему длины слова и сегментации: эти приложения были разработаны для телефонов с сенсорным экраном и рассчитаны на другой тип ввода — пользователю необходимо выбирать участок на экранной клавиатуре при помощи стилуса или пальца. Быстрый неотрывный ввод текста и полное доверие пользователя программе коррекции орфографии можно назвать новой целью программ такого рода: первым серьезным успехом такого рода можно назвать рекорд Гиннесса 2014 г. по самому быстрому набору текста, поставленный при помощи «Fleksy Keybord»[3].

На сегодняшний момент сформирован рынок программ, отвечающих за предсказывающий ввод текста: существует масса систем ускоренного ввода текста в цифровые устройства, где программное обеспечение в процессе набора предлагает варианты окончания слов и фраз, основываясь на словаре, а также может предлагать исправлять распространенные ошибки. Однако для русского языка существенным минусом оказывается неспособность существующих алгоритмов справиться с развитой морфологией — как минимум угадать верный падеж оказывается «дорогой» для реализации задачей. Стоит заметить, что для языков с агглютинативными чертами, таких как немецкий, в поздних версиях системы T9 предусмотрено деление слова на квазиморфемы, которые пользователь может набирать последовательно.

Технические детали и способы обучения, использовавшиеся в этих программах, представляют коммерческую тайну и не опубликованы, но можно предположить, что широко использовались как символьные n-граммы, так и n-граммы по словоформам, а в случае новейших версий этих программ, очевидно, и частеречная разметка, так как все эти системы

[3] Фраза *The razor-toothed piranhas of the genera Serrasalmus and Pygocentrus are the most ferocious freshwater fish in the world. In reality they seldom attack a human* была набрана за 18 секунд при средней объявленной разработчиками скорости на «Swype» в 30—40 слов в минуту. Более подробно см. электронный ресурс: http://www.guinnessworldrecords.com/news/2014/5/fastest-touch-screen-text-message-record-officially-broken-with-fleksy-keyboard-57380/.

поддерживают предсказывающее дописывание фраз и угадывание целевого слова по нескольким набранным буквам и предыдущему контексту. Новейшие программы, такие как недавно (в конце 2015 г.) вышедшая «Яндекс-клавиатура»[4], в общем отвечают следующим требованиям мобильного рынка: 1) большой словарь, 2) способность к обучению под пользователя: добавление новых слов, учет набранного текста, 3) учет контекста, 4) предложение продолжения фразы, 5) автоматическое переключение между языками (первая программа такого рода для русского языка — «Punto Switcher», появившаяся в 2001 г.[5]).

6. Контекстные методы, применимые для русского языка

Русский язык обладает рядом фонетических и морфонологических особенностей, представляющих интерес для занимающегося обработкой естественного языка исследователя, таких как небольшое количество склонений и развитая суффиксация, а также нестрогий порядок слов. Все это позволяет использовать различную информацию о контексте как при обнаружении опечаток, так и при их исправлении. Разберем более подробно, какую информацию для ранжирования кандидатов на исправление возможно использовать, когда опечатка уже найдена.

1. Фонетический контекст — характерно ли сочетание звуков в слове для данного языка. Сюда же относится информация о структуре слога. Обычно такая информация кодируется либо правилами, либо символьными n-граммами.

2. Морфологический контекст — характерна ли для языка наблюдаемая последовательность морфологических категорий, присутствует ли графическое выражение согласования и управления.

3. Синтаксический контекст — сведения о линейной последовательности синтаксических отношений используют для ранжирования кандидатов, что крайне важно для русского языка, где синтаксис не только позволяет выбрать правильного кандидата на исправление среди конкурирующих форм внутри парадигмы (3), но и дает существенное преимущество при обнаружении опечаток в некоторых моделях, построенных на принципе, при котором потенциально каждое слово может быть источником ошибки.

(3) *Велел проверить урокп → Велел проверить урок* или *Велел проверить уроки.*

4. Семантический контекст — сведения о сочетаемости слов друг с другом, общности их тематики, стилистические особенности. Используются, как правило, модели из дистрибутивной семантики, векторные модели и более простые методы нахождения частоты совместной встречаемости отдельных словоформ. При использовании принципа «каждое слово — потенциальный источник ошибки» такие контексты также способствуют не только правильному ранжированию кандидатов, но и обнаружению real-word ошибок (4).

(4) *Шел проливной вождь → Шел проливной дождь.*

Как будет видно из таблиц 3 и 4 в разделе 6.3, языковые модели, построенные на словоформах и частях речи, крайне полезны для работы системы, однако n-граммы по словоформам оказываются очень чувствительны к стилистическим и жанровым особенностям текста: если источник, из которого получены частоты совместной встречаемости, сильно отличается от материала, на котором модель будет работать, качество исправления падает.

6.1. Специфика электронных текстов

При подготовке системы автоматического корректора орфографии, применяемого на электронных текстах, важны следующие факторы:

[4] Электронный ресурс: https://yandex.ru/blog/company/klaviatura-na-vse-sluchai-zhizni.

[5] Электронный ресурс: https://yandex.ru/blog/company/39620/.

1. Если определение опечаток производится с опорой на словарь, то необходимо предусмотреть обработку несловарных вхождений, которыми чрезвычайно богаты тексты блогов и полный список которых невозможно учесть в словаре: топонимы, имена собственные, редкие фамилии, аббревиатуры и пр. В идеальной ситуации правильно создавать специальный газетир (gazetteer, инструмент, помогающий обнаружить имена собственные и объединяющий многословные имена собственные в цельную единицу на основании словарей и правил).

2. Бóльшая часть отклонений от словаря носит системный характер — типичные опечатки и орфографические ошибки хорошо описываются правилами. Для исправления орфографии в тексте на русском языке зачастую (например, при исправлении поисковых запросов[6]) первичное внимание уделяют правилам проверки безударных гласных и сочетаний согласных (*яблако → яблоко*, *лесница → лестница*). Такой модуль проверки несловарного слова обычно включается в систему ранжирования кандидатов и дает преимущество словарному кандидату, выводимому из наблюдаемого слова в соответствии с правилами. То же справедливо и для клавиатурных опечаток, однако количество правил, описывающих близость клавиш друг к другу, чрезмерно велико, поэтому для задания вероятностей символьных замен используют матрицу весов, определяемую эмпирически (при таком подходе опечатка *скахка → сказка* вероятнее, чем *скахка → скалка*, хотя расстояние Левенштейна в обоих случаях одинаковое).

Особенно хотелось бы отметить ошибки оптического распознавания в случае оцифрованных текстов: они также могут быть описаны корпусом правил, однако стоит учитывать изменение орфографической нормы с течением времени, см. пример из [Беликов 2006: 45—46]:

(5)	**Бумажное издание 1937 г.**	**ФЭБ, 2006 г.**
а.	*С кого начнет он? Всё равно:*	*С кого начнет он? Все равно:*
б.	*В большом рассеяньи взглянул,*	*В большом рассеянье взглянул,*
в.	*Ревнивый шопот модных жен*	*Ревнивый шепот модных жен*
г.	*И к шутке, с желчью пополам,* *И злости мрачных эпиграм.*	*И к шутке с желчью пополам* *И злости мрачных эпиграмм.*

При обучении на выборке возникает следующая дилемма: используя обучение с учителем, исследователь вряд ли сможет получить достаточную по объему выборку (например, в несколько миллионов слов), так как ручная разметка такого количества ошибок требует временных и материальных затрат, а качество свободно распространяемых программ исправления ошибок для этого будет явно недостаточным. Однако для применения контекстных методов зачастую требуется информация, которую можно получить и на неразмеченном корпусе: частотность словоформ, их сочетаемость и т. д., что позволяет применить в данном случае обучение без учителя.

Стоит обратить внимание и на другую проблему: исследователь, взявший большую выборку для обучения (например, несколько миллионов предложений), не застрахован от того, что получит непригодную для интерпретирования результатов выборку, не сбалансированную по таким важным параметрам, как тематика текстов, их происхождение, возраст и социальное положение автора. Это означает, что распределение опечаток в обучающей выборке и «типичных» интернет-текстах, к которым предполагается применять программу коррекции орфографии, не будет совпадать, что приведет, например, к неправильному выбору словарных кандидатов.

[6] См. о практике компании «Яндекс»: https://yandex.ru/company/press_releases/1998/01-01_00.

6.2. Использование информации о различных уровнях языка

При создании модели исправления опечаток возможно использование различной языковой информации, которую можно разделить на два типа: 1) применяемая при порождении, притом необязательно контекстная; 2) применяемая при ранжировании отобранных кандидатов, зачастую опирающаяся на контекст.

Информация, используемая при генерации кандидатов, в общем случае может быть а) фонетической — примером такого подхода служит алгоритм «Metaphone» [Philips 2000] для английского языка; аналогичный алгоритм для русского языка был впервые применен в [Sorokin, Shavrina 2016]; б) основанной на расстоянии Левенштейна — поиск происходит по словарным вхождениям на определенном расстоянии, на этом принципе основывается большинство программ коррекции орфографии для русского языка; в) основанной на клавиатурной близости или вручную написанных правилах — поиск по словарю осуществляется по некоторым заранее написанным правилам, способным исправлять частотные ошибки (коррекция орфографии запросов в «Яндексе»).

Информация, применяемая при ранжировании кандидатов, часто бывает а) морфологической — используются частеречные n-граммы, а также n-граммы с полной морфологической информацией (можно предполагать, что так действует «MS Word»), а также «yspell», «aspell» и «uspell»; б) синтаксической — таким образом при построении синтаксического дерева некоторые опечатки исправляет система «Abbyy compreno»[7]; в) семантической — для русского языка таких систем пока нет за исключением попыток в [Sorokin, Shavrina 2016], для английского языка первой была работа [Budanitsky, Hirst 2006] (для поиска словарных опечаток использовался «WordNet» [Fellbaum 1998]), позже подход был развит в [Schaback, Li 2007; Flor 2012], где статистическая информация о совместной встречаемости извлекалась из большого корпуса. В качестве меры близости использовалась позитивная нормализованная поточечная взаимная информация (PNPMI). В задачах выбора оптимального кандидата при поисковых запросах используется также статистика кликовых данных и история запросов пользователей [Иванов 2016].

6.3. SpellRuEval — первое соревнование по автоматическому исправлению опечаток в русских текстах

Автоматическая нормализация «грязных» текстов и коррекция орфографии в масштабе международных соревнований стали привлекать заслуженное внимание только в последнее десятилетие: это прежде всего первые соревнования «Helping Our Own» (HOO) 2011 и 2012 гг. [Dale, Kilgarriff 2011; Dale et al. 2012], а также «Microsoft Spelling Alteration Workshop» [Lueck 2011] для английского языка и «QALB» для арабского языка [Mohit et al. 2014; Rozovskaya et al. 2015].

На материале русского языка испытаний такого рода не проводилось до 2016 г., когда в рамках конференции «Диалог» было проведено первое соревнование по автоматическому исправлению опечаток «SpellRuEval»[8] [Sorokin et al. 2016]. Была получена базовая модель (baseline) и выявлены наиболее эффективные подходы к обработке социальных медиа (social media), а также были получены важные результаты относительно применимости контекстных методов на материале русского языка. Впервые были предприняты попытки учесть морфологическую и семантическую информацию при исправлении опечаток в русскоязычных интернет-текстах, а также применить методы дистрибутивной семантики в дополнение к традиционным n-граммным моделям, хотя большинство из них и не увенчалось успехом. Краткий конспект методологии участников и результаты соревнования и приводятся в таблицах 3 и 4 соответственно.

[7] Электронный ресурс: http://www.abbyy.ru/isearch/compreno/.

[8] Полная информация об итогах соревнования доступна по адресу: https://drive.google.com/drive/u/0/folders/0B8XxHuDfyogmMnM5RFdKdWdzZmM.

Таблица 3

Подход участников SpellRuEval к созданию программы-корректора орфографии для русского языка

Код группы	Методы	Число словоформ в словаре	Число лемм в словаре	Обнаружение ошибок	Исправление ошибок	Частотные ошибки	Дополнительная обучающая выборка
A	Расстояние Левенштейна для поиска кандидата и языковая модель для его отбора	3 700 000	230 000	Словарное на основе расстояния Левенштейна, смежности на клавиатуре и графической схожести	Триграммная языковая модель	Неправильное ранжирование кандидатов, синтаксические ошибки, словарные ошибки	Корпус из 213 млн слов с морфологической разметкой для языковой модели
B	Коррекция орфографии на нескольких уровнях, модель ошибок, n-граммы по словам и частям речи	3 700 000	230 000	Каждое слово потенциально содержит ошибку, лучший кандидат отбирается за счет коррекции	Расстояние Левенштейна и фонетические ключи для поиска кандидата, триграммная языковая модель и модель ошибок для ранжирования	Семантические ошибки, неправильное ранжирование кандидатов	50 млн слов без разметки из «Живого Журнала»
C	Word2Vec, n-граммы, гибридная модель ошибок: 1) традиционная channel model, 2) модель из работы [Brill, Moore 2000], 3) channel model с расширенным окном	Без словаря	Без словаря	Error detection confidence classifier, используется vector scores из Word2Vec	Классификатор слов с опечатками на основе Word2Vec	Неправильное ранжирование кандидатов, синтаксические ошибки, словарные ошибки	Корпус русских учебных текстов (КРУТ) [Зевахина, Джакупова 2015]: 2,5 млн слов + 13,8 млн слов из блогов + 22,2 млн слов из газет для языковой модели

Код группы	Методы	Число словоформ в словаре	Число лемм в словаре	Обнаружение ошибок	Исправление ошибок	Частотные ошибки	Дополнительная обучающая выборка
D	Расстояние Левенштейна, автоматическое конструирование парадигм, основанное на правилах	5000000	260000	Словарное + правила суффиксации	Расстояние Левенштейна; если кандидатов 2 и более, выбор между ними случаен	Неверные окончания, неправильное ранжирование кандидатов	Только данные соревнования
E	N-граммы, поиск по словарю на правилах	5095000	390000	Словарное (словарь OpenCorpora.org [Bocharov et al. 2013])	Ранжирование кандидатов с помощью триграмм обучающей выборки и расстояние Левенштейна	Неправильное ранжирование кандидатов, ошибки на разделение пробелом	400 млн слов из газет, социальных медиа и «Википедии»
F	Векторная модель, «Jflex», «Apache Tika»	5095000	390000	Словарное (словарь OpenCorpora.org [Bocharov et al. 2013])	Ранжирование синтаксических деревьев предложения с приоритетом у деревьев наибольшей длины, затем исправление словоформы на основании синтаксиса и частей речи	Пропущенные ошибки, орфографические ошибки	Только данные соревнования
G	N-граммы по словоформам и частям речи	5500000	400000	Словарное	Ранжирование кандидатов на основании триграмм, би- и триграмм частей речи и частот словоформ	Неправильное ранжирование кандидатов, ошибки на разделение пробелом	Корпус со снятой омонимией [НКРЯ]

Таблица 4

Итоги соревнования SpellRuEval

Место	Код группы	Точность	Полнота	F-мера	Аккуратность
1	B	81,98	69,25	75,07	70,32
2	G	67,54	62,31	64,82	61,35
3	A	71,99	52,31	60,59	58,42
4	E	60,77	50,75	55,31	55,93
BASELINE		55,91	46,41	50,72	48,06
5	C	74,87	27,99	40,75	50,45
6	D	23,50	30,00	26,36	24,95
7	F	17,50	9,65	12,44	33,96

По результатам соревнования получены важные практические выводы по следующим проблемам: 1) проблема словарей, 2) обучающая выборка, 3) использование контекстных методов для социальных медиа. Большинство участников не отказалось от использования словарей, и их объем в словоформах составляет в среднем более 3 млн. Оказалось, что критическим является не размер, а качество используемых словарей: полнота парадигм, отсутствие «мусора». Таким проблемам подвержены прежде всего открытые ресурсы. Перспектива возможного отказа от использования словарей и необходимых условий описана в разделе 7.

Обучающая выборка у участников, демонстрирующих наилучший результат, как показано в таблицах 2 и 3, была средних размеров — это корпуса текстов объемом более нескольких десятков миллионов слов, тогда как в упоминавшемся соревновании «Microsoft Spelling Alteration Workshop» на материале английского языка использовался корпус TREC[9], содержащий более 2 млрд поисковых запросов. Поскольку использование таких выборок подразумевает обучение без учителя, контекстная информация, извлекаемая из текстов, чаще всего ограничивается следующими данными: 1) символьные n-граммы (участники в основном использовали триграммы), 2) частота словоформ, 3) n-граммы частеречных меток.

7. Открытые вопросы, перспективы

В сфере автоматического исправления опечаток с учетом контекста в первую очередь остается открытым вопрос состава словарей: как видно из таблицы 3, большинство современных словарей имеет достаточно большой объем (более 3 млн словоформ), тогда как некоторые системы уже практикуют с относительным успехом полный отказ от словаря, как, например, система команды C, основанная на технологии [Whitelaw et al. 2009]. Потенциально возможным отказ от словаря делает развитие векторных технологий, таких как «Word2Vec» [Mikolov et al. 2013], «GloVe» [Pennington et al. 2014], «DSSM» [Huang et al. 2013], — в случае внедрения таких технологий разработчикам будут доступны более тонкие контекстные зависимости (прежде всего, семантические), находящиеся за пределами возможностей n-граммных моделей. Однако пока системы подобного рода не достигли большого успеха в исправлении опечаток.

С использованием векторных моделей связана и надежда на решение следующей проблемы: некоторые слова при большой графической близости имеют разную дистрибуцию, например, пара *телифоны* (отряд паукообразных) и *телефоны*, где формально оба слова являются словарными. При этом в предложении типа *Девушка, дайте телифон!* слово *телифон* является словарной (real-word) ошибкой и имеет нетипичный контекст.

[9] Text REtrieval Conference (TREC), 2011. Материалы доступны по адресу: http://trec.nist.gov/.

Одной из актуальных и проблемных с точки зрения контекстных методов сфер остаются короткие тексты, такие как тексты из социальной сети «Twitter», поскольку чем меньше текст, тем меньше информации мы можем извлечь из контекста и тем хуже будет выбор кандидата на исправление, к тому же сами тексты зачастую содержат много сокращений и искажений, нехарактерных для стандартной языковой модели. Для таких коротких текстов еще одним важным средством, работающим при помощи анализа символьного контекста, может стать так называемый infinity-gram [Shuyo 2012], открытый для свободного использования в начале 2016 г. Суть метода, разработанного специально для определения языка на текстах из «Twitter'а», но потенциально широко применимого в NLP, кратко сводится к следующему: 1) используется модель наибольшей подстроки (maximal substring model) или логистическая регрессия на n-граммах; 2) для обучения берется корпус текстов непосредственно из «Twitter'а» (или любого другого целевого ресурса), на котором строится языковая модель. Для русского языка такой алгоритм еще не применялся, так как использовался только корпус языков с латинской графикой.

Остается открытым вопрос о применении в контекстных моделях всевозможных тезаурусов — в качестве перспективного направления развития технологий исправления опечаток можно рассматривать аналогичные подходы на материале других языков, ср. недавно вышедшую систему «WNSpell» для английского языка [Huang 2016], работающую на базе тезауруса «WordNet» [Fellbaum 1998]. По наблюдению самого разработчика системы «WNSpell», внедрение формализованной семантики такого рода существенно помогло улучшить точность контекстного выбора кандидата на исправление, однако минусом оказалась зависимость качества работы системы от проработанности того или иного значения в самом тезаурусе.

СПИСОК ЛИТЕРАТУРЫ / REFERENCES

Байтин 2008 — Байтин А. Исправление поисковых запросов в Яндексе // Российские интернет-технологии, 2008. [Baytin A. Correction of search requests in Yandex. *Rossiyskie internet-tekhnologii*, 2008.]

Беликов 2006 — Беликов В. И. Оцифрованные тексты как материал для словаря русских регионализмов // Герд А., Захаров В., Митрофанова О. (ред.) Труды международной конференции «Корпусная лингвистика-2006». СПб.: Изд-во СПбГУ, 2006. С. 43—51. [Belikov V. I. Digitized texts as material for the dictionary of Russian regionalisms. *Trudy mezhdunarodnoy konferentsii «Korpusnaya lingvistika-2006»*. Gerd A., Zakharov V., Mitrofanova O. (eds.). St. Petersburg: St. Petersburg State Univ. Publ., 2006. Pp. 43—51.]

Зевахина, Джакупова 2015 — Зевахина Н. А., Джакупова С. С. Корпус русских учебных текстов: архитектура и перспективы // Селегей В. П. (ред.). *Компьютерная лингвистика и интеллектуальные технологии. По материалам Международной ежегодной конференции «Диалог»*. М.: РГГУ, 2015. [Corpus of Russian student texts: Design and prospects. *Komp'yuternaya lingvistika i intellektual'nye tekhnologii. Po materialam ezhegodnoi Mezhdunarodnoi konferentsii «Dialog»*. Selegey V. (ed.). Moscow: Russian State Univ. for the Humanities, 2015. Available at: http://www.dialog-21.ru/digests/dialog2015/materials/pdf/ZevakhinaNADzhakupovaSS.pdf.]

Иванов 2016 — Иванов Г. Использование кликовых данных для улучшения исправления опечаток // Yandex Data Fest, 2016. [Ivanov G. Use of click data for the improvement of debugging. *Yandex Data Fest*, 2016. Available at: https://events.yandex.ru/lib/talks/3991.]

Левенштейн 1965 — Левенштейн В. И. Двоичные коды с исправлением выпадений, вставок и замещений символов // Доклады АН СССР. 1965. Т. 163. № 4. С. 845—848. [Levenshtein V. I. Binary codes with correction of fallouts, insertions, and symbol substitutions. *Doklady AN SSSR*. 1965. Vol. 163. No. 4. Pp. 845—848.]

НКРЯ — Национальный корпус русского языка // http://www.ruscorpora.ru. [*Natsional'nyi korpus russkogo yazyka* [Russian National Corpus]. Available at: http://www.ruscorpora.ru.]

Осовцев 1999 — Осовцев С. Свобода опечаток: опечатки от Ромула до наших дней // Книжное обозрение. 1999. № 40. С. 6. [Osovtsev S. Freedom of misprints: Misprints from Romulus till our time. *Knizhnoe obozrenie*. 1999. No. 40. P. 6.]

Ушаков — Ушаков Д. Н. (ред.). Толковый словарь русского языка. М.: Государственное издательство иностранных и национальных словарей, 1935—1940. [Ushakov D. N. (ed.). Tolkovyi slovar' russkogo

yazyka [Defining dictionary of the Russian language]. Moscow: Gosudarstvennoe izdatel'stvo inostrannykh i natsional'nykh slovarei, 1935—1940.]

Шаврина, Сорокин 2015 — Шаврина Т. О., Сорокин А. А. Моделирование расширенной лемматизации для русского языка на основе морфологического парсера TnT-Russian // Селегей В. П. (ред.). *Компьютерная лингвистика и интеллектуальные технологии. По материалам Международной ежегодной конференции «Диалог»*. [Modeling advanced lemmatization for Russian language using TnT-Russian morphological parser. *Komp'yuternaya lingvistika i intellektual'nye tekhnologii. Po materialam ezhegodnoi Mezhdunarodnoi konferentsii «Dialog»*. Selegey V. (ed.). Moscow: Russian State Univ. for the Humanities, 2015. Available at: http://www.dialog-21.ru/digests/dialog2015/materials/pdf/ShavrinaTOSorokinAA.pdf.]

Шерих 2004 — Шерих Д. Ю. «А» упало, «Б» пропало… Занимательная история опечаток. М.: МиМ-Дельта, 2004. [Sherikh D. Yu. *«A» upalo, «B» propalo… Zanimatel'naya istoriya opechatok* ["A" fell down, "B" disappeared… An entertaining theory of misprints]. Moscow: MiM-Del'ta, 2004.]

Bahl et al. 1989 — Bahl L. R., Brown P. F., Desouza P. V., Mercer R. L. A tree-based statistical language model for natural language speech recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*. 1989. Vol. 37. No. 7. Pp. 1001—1008.

BNC — *The British National Corpus*, version 3 (BNC XML Edition). Distributed by Oxford Univ. Computing Services on behalf of the BNC Consortium, 2007. Available at: http://www.natcorp.ox.ac.uk/.

Bocharov et al. 2013 — Bocharov V. V., Alexeeva S. V., Granovsky D. V., Protopopova E. V., Stepanova M. E., Surikov A. V. Crowdsourcing morphological annotation. *Komp'yuternaya lingvistika i intellektual'nye tekhnologii. Po materialam ezhegodnoi Mezhdunarodnoi konferentsii «Dialog»*. Vol. 12(19). Selegey V. (ed.). Moscow: Russian State Univ. for the Humanities, 2013. Pp. 109—124.

Brill, Moore 2000 — Brill E., Moore R. C. An improved error model for noisy channel spelling correction. *Proceedings of the 38th Annual meeting of Association for Computational Linguistics*. Hitoshi I. (ed.). Hong Kong: Association for Computational Linguistics, 2000. Pp. 286—293.

Brown et al. 1990 — Brown P., Cocke J., Della Pietra S., Della Pietra V., Jelinek F., Mercer R., Roosin P. A statistical approach to machine translation. *Computational Linguistics*. 1990. Vol. 16. No. 2. Pp. 79—85.

Budanitsky, Hirst 2006 — Budanitsky A., Hirst G. Evaluating WordNet-based measures of semantic distance. *Computational Linguistics*. 2006. Vol. 32. No. 1. Pp. 13—47.

Burr 1987 — Burr D. J. Experiments with a connectionist text reader. *Proceedings of the First international conference on neural networks*. Vol. 4. Caudill M., Butler C. (eds.). San Diego (CA): SOS Printing, 1987. Pp. 717—724.

Chen, Goodman 1998 — Chen F. S., Goodman J. *An empirical study of smoothing techniques for language modeling*. Technical Report TR-10-98. Cambridge (MA): Harvard Univ., 1998.

Choueka 1988 — Choueka Y. Looking for needles in a haystack, or Locating interesting collocational expressions in large textual databases. *Proceedings of RIAO Conference on user-oriented content-based text and image handling*. Cambridge (MA): MIT, 1988. Pp. 609—623.

Church 1988 — Church K. W. A stochastic parts program and noun phrase parser for unrestricted text. *Proceedings of the 2nd Applied natural language processing conference*. Bates M. et al. (eds.). Austin (Texas): Association for Computational Linguistics, 1988. Pp. 136—143.

Dale, Kilgarriff 2011 — Dale R., Kilgarriff A. Helping our own: The HOO 2011 pilot shared task. *Proceedings of the 13th European workshop on natural language generation*. Gardent C., Striegnitz K. (eds.). Nancy: Association for Computational Linguistics, 2011. Pp. 242—250.

Dale et al. 2012 — Dale R., Anisimoff I., Narroway G. A report on the preposition and determiner error correction shared task. *Proceedings of the NAACL Workshop on innovative use of NLP for building educational applications*. Tetreault J., Burstein J., Leacock C. (eds.). Montreal: Association for Computational Linguistics, 2012. Pp. 216—224.

Damerau 1964 — Damerau F. J. A technique for computer detection and correction of spelling errors. *Communications of the Association for Computing Machinery*. 1964. Vol. 7. No. 3. Pp. 171—176.

Deffner et al. 1990 — Deffner R., Eder K., Geiger H. Word recognition as a first step towards natural language processing with artificial neural nets. *Proceedings of KONNAI-90*. 1990. Vol. 252. Pp. 221—225.

Fellbaum 1998 — Fellbaum C. *WordNet: An electronic lexical database*. Cambridge (MA): MIT Press, 1998.

Flor 2012 — Flor M. Four types of context for automatic spelling correction. *Traitement Automatique des Langues*. 2012. Vol. 53. No. 3. Pp. 61—99.

Garside et al. 1987 — Garside R., Leach G., Sampson G. *The computational analysis of English: A corpus-based approach*. New York: Longman, 1987.

Gersho, Reiter 1990 — Gersho M., Reiter R. Information retrieval using self-organizing and heteroassociative supervised neural networks. *Proceedings of International joint conference on neural networks (IJCNN-'90)*. Vol. 2. San Diego (CA), 1990. Pp. 361—364.

Han et al. 2012 — Han B., Cook P., Baldwin T. Lexical normalization of short text messages. *Proceedings of the 49th Annual meeting of the association for computational linguistics: Human language technologies*. Vol. 1. Portland (OR): Association for Computational Linguistics. Pp. 368—378.

Hanson et al. 1976 — Hanson A. R., Riseman E. M., Fisher E. Context in word recognition. *Pattern Recognition*. 1976. Vol. 8. Pp. 35—45.

Heidorn et al. 1982 — Heidorn G. E., Jensen K., Miller L. A., Byrd R. J., Chodorow M. S. The EPISTLE text-critiquing system. *IBM Systems Journal*. 1982. Vol. 21. No. 3. Pp. 305—326.

Huang 2016 — Huang B. WNSpell: A WordNet-based spell corrector. Paper presented at Global WordNet Conference, 2016.

Huang et al. 2013 — Huang P., He X., Gao J., Deng L. Learning deep structured semantic models for web search using clickthrough data. Paper presented at International conference on information and knowledge management, 2013. Available at: https://www.microsoft.com/en-us/research/publication/learning-deep-structured-semantic-models-for-web-search-using-clickthrough-data/

Jelinek et al. 1991 — Jelinek F., Merialdo B., Roukos S., Strauss M. A dynamic language model for speech recognition. *Proceedings of the DARPA speech and natural language workshop.* Price P. (ed.). Pacific Grove (CA): Association for Computational Linguistics, 1991. Pp. 293—295.

Kempen, Vosse 1990 — Kempen G., Vosse T. A language sensitive text editor for Dutch. *Proceedings of the Computers and writing III conference*. O'Brian H., Williams N. (eds.). Dordrecht: Kluwer, 1990. Pp. 68—77.

Kukich 1990 — Kukich K. A comparison of some novel and traditional lexical distance metrics for spelling correction. *Proceedings of the International neural network conference*. Vol. 2. Paris: Springer Science & Business Media, 1990. Pp. 309—313.

Kukich 1992 — Kukich K. Techniques for automatically correcting words in text. *ACM Computing Surveys*. 1992. Vol. 24. No. 4. Pp. 377—439.

Lueck 2011 — Lueck G. A data-driven approach for correcting search queries. Paper presented at Spelling alteration for web search workshop, 2011. Available at: https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/en-us-events-spellerworkshop2011-spelling_alteration_workshop.pdf

McEnery, Hardie 2011 — McEnery T., Hardie A. *Corpus linguistics: Method, theory and practice.* Cambridge: Cambridge Univ. Press, 2011.

Mikolov et al. 2013 — Mikolov T., Chen K., Corrado G., Dean J. *Efficient estimation of word representations in vector space*. Ms., arXiv preprint arXiv:1301.3781. Available at: https://arxiv.org/abs/1301.3781.

Mitton 1987 — Mitton R. Spelling checkers, spelling correctors, and the misspellings of poor spellers. *Information Processing & Management.* 1987. Vol. 23. No. 5. Pp. 495—505.

Mohit et al. 2014 — Mohit B., Rozovskaya A., Habash N., Zaghouani W., Obeid O. The first QALB shared task on automatic text correction for Arabic. *Proceedings of EMNLP Workshop on Arabic natural language processing.* Habash N., Vogel S. (eds.). Doha: Curran Associates, 2014. Pp. 39—48.

Norvig 2010 — Norvig P. *How to write a spelling corrector*. Ms., 2010. Available at: http://norvig.com/spell-correct.html.

Oshika et al. 1988 — Oshika T., Machi F., Evans B., Tom J. Computational techniques for improved name search. *Proceedings of the 2nd Applied natural language processing conference*. Bates M. et al. (eds.). Austin (Texas): Association for Computational Linguistics, 1988. Pp. 203—210.

Pennington et al. 2014 — Pennington J., Socher R., Manning C. D. Glove: Global vectors for word representation. *Proceedings of the Empirical methods in natural language processing (EMNLP 2014)*. Marton Y. (ed.). Doha: Association for Computational Linguistics, 2014. Pp. 1532—1544.

Peterson 1986 — Peterson J. L. A note on undetected typing errors. *Communications of the ACM*. 1986. Vol. 29. No. 7. Pp. 633—637.

Philips 2000 — Philips L. The double metaphone search algorithm. *C/C++ Users Journal*. 2000. Vol. 18. No. 6. Pp. 38—43.

Popescu, Vo 2014 — Popescu O., Vo N. P. A. Fast and accurate misspelling correction in large corpora. *Proceedings of the Empirical methods in natural language processing (EMNLP 2014)*. Marton Y. (ed.). Doha: Association for Computational Linguistics, 2014. Pp. 1634—1643.

Richardson, Braden-Harder 1988 — Richardson S. D., Braden-Harder L. C. The experience of developing a larger scale natural language text processing system: CRITIQUE. *Proceedings of the 2nd Applied natural language processing conference*. Bates M. et al. (eds.). Austin (Texas): Association for Computational Linguistics, 1988. Pp. 195—202.

Rozovskaya et al. 2015 — Rozovskaya A., Bouamor H., Habash N., Zaghouani W., Obeid O., Mohit B. The second QALB shared task on automatic text correction for Arabic. *Proceedings of the Second workshop on Arabic natural language processing*. Tomeh N., Bouamor H. (eds.). Beijing: Association for Computational Linguistics, 2015. Pp. 26—35.

Rumelhart et al. 1986 — Rumelhart D. E., Hinton G. E., Williams R. J. Learning internal representations by error propagation. *Parallel distributed processing: Explorations in the microstructure of cognition*. Rumelhart D. E., McClelland J. L., Bradford E. (eds.). Cambridge (MA): MIT Press, 1986. Pp. 318—362.

Schaback, Li 2007 — Schaback J., Li F. Multi-level feature extraction for spelling correction. Paper presented at IJCAI-2007 Workshop on analytics for noisy unstructured text data, 2007. Available at: http://research.ihost.com/and2007/.

Schäfer, Bildhauer 2013 — Schäfer R., Bildhauer F. Web corpus construction. *Synthesis Lectures on Human Language Technologies*. 2013. Vol. 6. No. 4. Pp. 1—145.

Shuyo 2012 — Shuyo N. Short text language detection with infinity-gram. Paper presented at Naist seminar, 2012. Available at: http://www.slideshare.net/shuyo/short-text-language-detection-with-infinity-gram-12949447.

Smadja 1991 — Smadja F. *Extracting collocations from text. An application: Text generation*. PhD dissertation. New York: Columbia Univ., 1991.

Sorokin et al. 2016 — Sorokin A. A., Baitin A. V., Galinskaya I. E., Shavrina T. O. SpellRuEval: The first competition on automatic spelling correction for Russian. *Komp'yuternaya lingvistika i intellektual'nye tekhnologii. Po materialam ezhegodnoi Mezhdunarodnoi konferentsii* «*Dialog*». Selegey V. (ed.). Moscow: Russian State Univ. for the Humanities, 2016. Pp. 660—674.

Sorokin, Shavrina 2016 — Sorokin A. A., Shavrina T. O. Automatic spelling correction for Russian social media texts. *Komp'yuternaya lingvistika i intellektual'nye tekhnologii. Po materialam ezhegodnoi Mezhdunarodnoi konferentsii* «*Dialog*». Selegey V. (ed.). Moscow: Russian State Univ. for the Humanities, 2016. Pp. 688—702.

Toutanova, Moore 2002 — Toutanova K., Moore R. C. Pronunciation modeling for improved spelling correction. *Proceedings of the 40th Annual meeting on Association for Computational Linguistics.* Pierre I. (ed.). Philadelphia: Association for Computational Linguistics, 2002. Pp. 144—151.

Van Berkel, DeSmedt 1988 — Van Berkel B., DeSmedt K. Triphone analysis: A combined method for the correction of orthographical and typographical errors. *Proceedings of the 2nd Applied natural language processing conference*. Bates M. et al. (eds.). Austin (Texas): Association for Computational Linguistics, 1988. Pp.77—83.

Whitelaw et al. 2009 — Whitelaw C., Hutchinson B., Chung G. Y., Ellis G. Using the web for language independent spellchecking and autocorrection. *EMNLP '09 Proceedings of the 2009 Conference on empirical methods in natural language processing*. Vol. 2. Singapore: Association for Computational Linguistics, 2009. Pp. 890—899.

Zamora et al. 1981 — Zamora E. M., Pollock J. J., Zamora A. The use of trigram analysis for spelling error detection. *Information Processing & Management*. 1981. Vol. 17. No. 6. Pp. 305—316.

Zipf 1949 — Zipf G. K. *Human behavior and the principle of least effort*. Cambridge (MA): Addison-Wesley Press, 1949.

Получено/received 17.01.2016 *Принято/accepted 07.02.2017*