П

УДК 004.89:004.93

Я. С. Пикалёв, А. С. Вовнянко, И. Я. Денищенко
Государственное учреждение «Институт проблем искусственного интеллекта», г. Донецк
83048, г. Донецк, ул. Артема, 118-б

АНАЛИЗ АВТОМАТИЧЕСКИХ СИСТЕМ ПРОВЕРКИ ПРАВОПИСАНИЯ РУССКОГО ЯЗЫКА

Ya. S. Pikalyov, A. S. Vovnyanko, I. Ya. Denishenko
Public institution «Institute of Problems of Artificial intelligence», c. Donetsk
83048, Donetsk, Artema str., 118-b

ANALYSIS OF AUTOMATIC VERIFICATION SYSTEMS SPELLING OF THE RUSSIAN LANGUAGE

Я. С. Пікальов, О. С. Вовнянко, І. Я. Денищенко
Державна установа «Інститут проблем штучного інтелекту», м. Донецьк
83048, м. Донецьк, вул. Артема, 118-б

АНАЛІЗ АВТОМАТИЧНИХ СИСТЕМ ПЕРЕВІРКИ ПРАВОПИСУ РОСІЙСЬКОЇ МОВИ

Данная работа посвящена исследованию автоматических систем проверки правописания русского языка. В работе рассматривается методика оценивания автоматических систем проверки правописания, разработанная авторами. Даётся описание отобранным системам. На основе проведенного исследования авторами выделяется наилучшая система, а также делается вывод о дальнейшей актуальности проведения разработок в данной области.
Ключевые слова: обработка естественного языка; автоматические системы проверки правописания русского языка; методика оценивания; метод анализа иерархий.

This work is devoted to the research of automatic spelling checkers of the Russian language. The technique of estimation of automatic spelling check systems developed by the authors is considered. A description is given to the selected systems. On the basis of the research carried out by the authors, the best system is singled out, and a conclusion is made about the further relevance of the development in this field.
Keywords: natural language processing; automatic spelling systems for the Russian language; method of assessment; method of analyzing hierarchies.

Дана робота посвячена дослідженню автоматичних систем перевірки правопису російської мови. В роботі розглядається методика оцінювання автоматичних систем перевірки правопису, розроблена авторами. Дається опис відібраним системам. На основі проведеного дослідження авторами виділяється найкраща система, а також робиться висновок про подальшу актуальність проведення розробок у цій галузі.
Ключові слова: обробка природної мови; автоматичні системи перевірки правопису російської мови; методика оцінювання; метод аналізу ієрархій.

Введение

Разработка систем проверки правописания, показывающих оптимальный результат, является одной из ключевых проблем направления обработки естественного языка (natural language processing, NLP). Актуальность данной проблемы подтверждается широким внедрением систем проверки правописания в большинство видов программных продуктов (текстовые редакторы, системы автоматического перевода и т.п.), а также использованием исследователями и разработчиками для других задач NLP и искусственного интеллекта. На момент написания данной работы в открытой печати отсутствуют статьи, посвященные задаче обзора систем проверки правописания для русского языка. В связи с этим, актуальной задачей является анализ систем проверки правописания для русского языка среди существующих решений.

Объектом настоящего исследования являются системы проверки правописания для русского языка. Предметом исследования является методика оценки системы проверки правописания для русского языка. **Цель работы** – выбор и анализ системы проверки правописания для русского языка.

Задачами работы являются:

1. Характеристика систем проверки правописания для русского языка для дальнейшей оценки.

2. Оценка систем проверки правописания для русского языка по разработанной авторской методике.

Методы исследования: метод анализа иерархий (МАИ) [1], метод Борда [2].

1 Обзор существующих аналогов

Так как существует большое множество различных решений среди автоматических систем проверки правописания (АСПП) для русского языка, был сформирован ряд критериев для их отбора:

– возможность прямого обращения (через web-интерфейс или в качестве программного модуля);

– возможность обратного получения списка возможных вариантов (некоторые решения просто подчеркивают слова);

– кроссплатформенность;

– свободный или условно-свободный доступ;

– возможность пословной проверки (большинство решений предназначены исключительно для проверки текстовых корпусов).

Из множества решений были отобраны следующие АСПП.

а) Яндекс.Спеллер [3] предназначен для проверки и исправления орфографических ошибок в русском, украинском, а также английском текстах. Пунктуационные, грамматические (ошибки согласования слов) и стилистические ошибки не исправляются. Данное решение использует правила орфографии и лексики современного языка, а также технологию машинного обучения (библиотеку CatBoost), которая обеспечивает исправление таких слов, как «адникасниеи» («одноклассники») и учитывает контекст («скучать музыку» – «скачать музыку»). Кроме того, Яндекс.Спеллер использует алгоритм для проверки внесловарных слов. Языковые модели состоят из сотен миллионов слов и словосочетаний. В качестве словарей используются словарные базы ОРФО. Данное решение использует web-интерфейс с собственным API (программный интерфейс приложения). У данного сервиса есть ограничение по количеству запросов: 10 тыс. обращений в сутки; и по количеству текстового объёма: 10 млн символов в сутки. Причём объём и количество запросов Яндекс может изменять, а также вправе отказать в доступе пользователю.

б) Google Spellchecker [4] имела собственное API до середины 2017 г. На текущий момент входит в состав продуктов Google таких, как Google Mail, Google Docs, Google Search. Данное решение основано на методах машинного обучения, а также на правилах орфографии и лексики современного языка; словарь состоит из более 1 млн слов. Ограничение по запросам как у Яндекс.Спеллер, 10 тыс. запросов, не более 1 млн символов, с последующей тарификацией: $20 за 1 млн символов. Google Spellcheker как и Яндекс.Спеллер предоставляет web-интерфейс для обработки клиентских запросов.

в) Hunspell [5] является бесплатной АСПП. Она основана на правилах орфографии и лексики современного языка, используя в качестве словарной базы словари MySpell, размером около 200 тыс. слов. В Hunspell помимо функции проверки ошибок используется морфологический анализатор. Данная АСПП нашла своё применение в таких решениях, как LibreOffice, OpenOffice, Mozilla Firefox, Thunderbird, Google Chrome, Opera и т.д. Главными преимуществами Hunspell являются: поддержка кодировки unicode; использование морфологии; применение метода n-gram для определения правильности написания слов; правила определения верного варианта написания основаны на транскрипциях; использование языка программирования C++. Стоит отметить, что Hunspell может использоваться как инструмент библиотекой проверки правописания PyEnchant [6], которая реализована на языке программирования Python.

г) LanguageTool [7] является условно-бесплатной АСПП, цена составляет $5 в месяц. В данном решении имеется поддержка как web-интерфейсов, так и программных библиотек; используется язык программирования Java. Размер словаря составляет около 300 тыс. слов. Стоит отметить, что система ориентирована в основном на английский и немецкий языки. Для этих языков предусмотрены нейросетевые алгоритмы, а также n-граммные языковые модели, которые отсутствуют для русского языка. Для него используется набор правил орфографии и лексики современного языка. Language Tool может использоваться в качестве расширения для Firefox, Chrome, Google Docs.

2 Экспертная оценка автоматических систем проверки правописания для русского языка

Для экспертной оценки АСПП был определён список критериев: интерфейс (K1); алгоритмы (K2); доступность (K3); размер словаря (K4); наличие документации (K5); орфографическая точность (ОТ) для слов с ошибками в начале слов (K6); ОТ для слов с ошибками в середине слов (K7); орфографическая точность для слов с ошибками в конце слов (K8); общая ОТ (K9).

ОТ является коэффициентом точности определения ошибок в слове с последующей заменой слова на корректное; введён авторами работы. Алгоритм определения ОТ в данной работе состоит в следующем

Взять n случайных слов из словаря (в данной работе n=1000), размером не менее 100 тыс. слов .

1. Провести зашумление слов, что подразумевает явное добавление ошибок в слова. Используется три категории зашумления слов: в начале слов; в середине слов; в конце слов. Пример зашумления для слова «лютеранка»: «яютеранка» – «лютреанка» – «лютерана».
2. Получив три массива зашумлённых слов, подать их в качестве входных значений выбранным системам проверки правописания, и получить результирующие значения в качестве векторов массивов исправленных слов. Пример исправления слова «лютерана» системой Hunspell: «лютеранка»; «лютеранина».
3. Провести оценку полученных векторов исправленных слов. Если в массиве возможных верных вариантов слова, эталонному слову соответствует первое слово в

массиве – то данной системе начисляется 1 балл. Если верный вариант находится в массиве, но оно не первое – то начисляется 0.5 балла. Если в массиве возможных вариантов слов отсутствует верное – то системе начисляется 0 баллов.

4. На основе полученных данных вычислить коэффициент ОТ для каждого уровня зашумлённости, используя формулу:

$$OT = \frac{\sum_{i=0}^{n} m_i}{n}; 0 \le n \le s_{voc}, \quad (1)$$

где n – это общее количество случайных слов; $\sum_{i=0}^{n} m_i$ – сумма полученных баллов для системы; s_{voc} – размер словаря, из которого случайным образом выбираются слова.

5. После получения коэффициентов вывести общий коэффициент ОТ (OT_{av}) на основе полученных на предыдущем этапе коэффициентов ОТ для каждой категории зашумления:

$$OT_{av} = \frac{OT_{beg} + OT_{mid} + OT_{end}}{3}, \quad (2)$$

где OT_{beg} – ОТ для зашумления в начале слов; OT_{mid} – для зашумления в середине слов; OT_{end} – для зашумления в конце слов.

Результаты вычисления коэффициентов ОТ по вышеуказанному алгоритму отображены в табл. 1.

Таблица 1 – Результаты вычисления коэффициетов ОТ для анализируемых АСПП

АСПП	OT_{beg}	OT_{mid}	OT_{end}	OT_{av}
Яндекс.Спеллер	0.56	0.81	0.64	0.67
Google Spellcheck	0.55	0.62	0.41	0.53
Hunspell	0.74	0.97	0.6	0.77
LanguageTool	0.84	0.83	0.67	0.78

Используя статистические методы (МАИ и метод Борда), а также выбранный список критериев и результаты из табл. 1, выведем наилучшие варианты для Яндекс.Спеллер(В1), Google Spellchecker (В2), Hunspell (В3), LanguageTool (В4). Стоит отметить, что для МАИ была использована собственная шкала распределения баллов — от 1/5 до 5, т.е. 5-балльная. Проведём оценку АСПП для МАИ (табл. 3) и метода Борда (табл. 4) на основе табл.2.

Таблица 2 – Оценочная таблица для критериев АСПП.

Оценка	К1	К2	К3	К4	К5	К6-К9
5	консольный и web	нейронные сети	бесплатный	полная	>500000	>0.9
4	консольный	машинное обучение	условно-бесплатный		200000...500000	0.7...0.9
3	web	набор лексических правил	платный	неполная	<200000	0.5...07
1				нет		<0.5

Таблица 3 – Результаты экспертной оценки в баллах для АСПП на основе МАИ

СПП	K1	K2	K3	K4	K5	K6	K7	K8	K9
Яндекс.Спеллер	3	3	3	5	5	3	5	3	3
Google Spellcheck	3	5	3	5	1	3	3	1	3
Hunspell	5	1	5	1	5	4	5	4	4
LanguageTool	5	1	3	1	5	5	5	4	4

После получения результатов экспертной оценки (табл. 3, 4) и определения локальных (для каждого критерия) и глобальных коэффициентов (для каждой АСПП) вычисляем: глобальный критерий ($v'A_j$) [1-2], наилучший вариант выбирается по максимальному значению; величину Борда (S_k) [2], наилучший вариант выбирается по минимальному значению (табл. 5).

Таблица 4 – Результаты экспертной оценки в баллах для АСПП на основе метода Борда

АСПП	K1	K2	K3	K4	K5	K6	K7	K8	K9
Яндекс.Спеллер	2	2	2	2	1	3	1	2	2
Google Spellcheck	2	1	2	1	2	3	2	3	2
Hunspell	1	3	1	2	1	2	1	1	1
LanguageTool	1	3	2	2	1	1	1	1	1

Таблица 5 – Итоговая таблица экспертной оценки для АСПП

	Яндекс.Спеллер	Google Spellcheck	Hunspell (PyEnchant)	Language Toolls
$v'A_j$	0.247	0.21	0.269	0.274
S_k	17	18	13	13

Дополнительно АСПП были проанализированы по критериям, взятых из методики оценок систем автоматического распознавания речи [8]. В данном случае для оценивания не применяется общеизвестная метрика WER (количество неправильно распознанных слов, Word Error Rate), т.к. она является лишь количественным показателем точности распознавания, т.е. показатель WER не ограничивается интервалом вероятности [0; 1] и не имеет верхнего предела. Например, слово, состоящее из 10 символов, может быть заменено на слово с 12 символами; в этом случае WER=120%, и, следовательно, показатель точности отрицательный (-20%). Поэтому для избежания подобных проблем используется модифицированный показатель ошибки распознавания соответствий (MER, Match Error Rate), основанный на величине относительной потери информации. Метод определения показателя MER подобно WER состоит в выравнивании двух текстовых строк (первая – это исправленное слово, а вторая – исходное слово) с помощью алгоритма динамического программирования с вычислением расстояния Левенштейна [9]. Расстояние Левенштейна – это метрика для определения схожести двух строк по минимальному количеству операций замены (S), удаления (D) и вставки (I) символов. Помимо этого MER оперирует количеством верно исправленных слов (H):

$$MER = \frac{S + D + I}{H + S + D + I}. \quad (3)$$

Результаты определения MER указаны в табл. 6.

Таблица 6 – Результаты вычисления коэффициентов MER для анализируемых АСПП

АСПП	MER_{beg}	MER_{mid}	MER_{end}	MER_{av}
Яндекс.Спеллер	0.89	0.767	0.58	0.75
Google Spellcheck	0.56	0.74	0.36	0.55
Hunspell	0.37	0.72	0.043	0.378
LanguageTool	0.27	0.787	0.37	0.478

Используя данные из табл. 5 и табл. 6, получаем следующее ранжирование:

1. МАИ: B4>B3>B1>B2;
2. Метод Борда: B4, B3>B1>B2;
3. MER: B3>B4>B2>B1.

Выводы

Таким образом, наилучший результат показывают Hunspell и LanguageTool. Наихудший результат показывают Яндекс.Спеллер и Google Spellcheck. Основываясь на полученных ранее данных, можно сделать вывод, что такой результат Яндекс.Спеллер и Google Spellcheker обоснован использованием больших словарей, в то время как Hunspell и Language Tools используют словари средних объёмов.

LanguageTool показывает наилучший результат для исправлений слов с ошибками в начале и в конце слов. Однако, по показателям исправления слов в середине, Hunspell показывает оптимальный результат ($OT_{mid} \geq 0.9$). Стоит отметить, что LanguageTool ориентирован на английский, немецкий и испанский языки, используя нейросетевые методы на основе n-gram, а для русского языка этого функционала нет.

На основе полученных результатов, можно сделать вывод, что ни один из проанализированных АСПП для русского языка не показал оптимальный результат, что делает актуальной задачу разработки подобной системы.

Список литературы

1. Саати Т. Принятие решений. Метод анализа иерархий : пер. с англ. [Текст] / Т. Саати. – М. : Радио и связь, 1993. – 278 с.
2. Кичмаренко О. Д. Теория принятия решений Раздел : Теория голосования. Методическое пособие [Электронный ресурс] / О. Д. Кичмаренко, А. П. Огуленко // ОНУ имени И.И. Мечникова. – 2012 – Режим доступа: http://ouek.onu.edu.ua/uploads/courses/smpr/Методичка_Теория_голосования.pdf. – (Дата обращения: 05.02.2014).
3. Спеллер – Технологии Яндекса [Электронный ресурс]. – Режим доступа: https://tech.yandex.ru/speller/. – (Дата обращения: 12.06.2018).
4. Using the web for language independent spellchecking and autocorrection [Текст] / Whitelaw C, Hutchinson B, Chung GY, Ellis G. // Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2 2009 Aug 6 (pp. 890-899). Association for Computational Linguistics.
5. Hunspell [Электронный ресурс]. – Режим доступа : http://hunspell.github.io/ – (Дата обращения: 25.05.2018).
6. Spellchecking library for python [Электронный ресурс]. – Режим доступа: https://github.com/rfk/pyenchant – (Дата обращения: 01.06.2018).

7. LanguageTool – Style and Grammar Checker [Электронный ресурс]. – Режим доступа: https://languagetool.org/ – (Дата обращения: 04.06.2018).
8. Карпов А. А. Методология оценивания работы систем автоматического распознавания речи [Текст] / Карпов Алексей Анатольевич, Кипяткова Ирина Сергеевна // Приборостроение. – 2012. – № 11. – URL: https://cyberleninka.ru/article/n/metodologiya-otsenivaniya-raboty-sistem-avtomaticheskogo-raspoznavaniya-rechi (дата обращения: 25.06.2018).
9. Levenshtein V. I. Binary codes capable of correcting deletions, insertions and reversals [Текст] / V. I. Levenshtein // Sov. Phys. Dokl. – 1966. – Vol. 6. – P. 707–710.

References

1. Saati T. *Prinyatiye resheniy. Metod analiza iyerarkhiy* : per. s angl. [Decision-making. Method for analyzing hierarchies: Per. with English], M. : Radio and Communication, 1993, 278 p.
2. Kichmarenko O. D. Ogulenko A. P. Teoriya prinyatiya resheniy Razdel : Teoriya golosovaniya. Metodicheskoye posobiye [Theory of decision-making Section: The theory of voting. Methodical manual]. *ONU imeni I.I. Mechnikova* [ONU named after II. Mechnikov], 2012, Access mode: http://ouek.onu.edu.ua/uploads/courses/smpr/Methodical_Theory_voting.pdf.
3. *Speller – Tekhnologii Yandeksa* [Speller - Yandex Technologies]. Access mode: https://tech.yandex.ru/speller/. - (Date of circulation: 12.06.2018).
4. Whitelaw C, Hutchinson B, Chung GY, Ellis G. Using the web for language independent spellchecking and autocorrection. *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*: Volume 2-Volume 2 2009 Aug 6 (pp. 890-899). Association for Computational Linguistics.
5. *Hunspell* [Electronic resource]. Access mode: http://hunspell.github.io/
6. *Spellchecking library for python* [Electronic resource]. Access mode: https://github.com/rfk/pyenchant -
7. *LanguageTool* Style and Grammar Checker [Electronic resource]. Access mode: https://languagetool.org/
8. Karpov Alexei Anatolyevich, Kipyatkova Irina Sergeevna Metodologiya otsenivaniya raboty sistem avtomaticheskogo raspoznavaniya rechi [Methodology for evaluating the operation of automatic speech recognition systems]. *Priborostroyeniye* [Instrument making], 2012, No. 11, URL: https://cyberleninka.ru/article/n/metodologiya-otsenivaniya-raboty-sistem-avtomaticheskogo-raspoznavaniya-rechi (date of circulation: June 25, 2013).
9. Levenshtein V. I. [Binary codes capable of correcting deletions, insertions and reversals]. *Sov. Phys. Dokl*, 1966. Vol. 6, pp. 707–710.

RESUME

Y. S. Pikaliov, A. S. Vovnyanko, I. Y. Denischenko

Analysis of automatic verification systems spelling of the Russian language

Background: the development of spelling systems that show the optimal result is one of the key problems in the direction of natural language processing (NLP). The relevance of this problem is confirmed by the extensive introduction of spelling systems in most types of software products (text editors, automatic translation systems, etc.), as well as using researchers and developers for other NLP tasks and artificial intelligence. There are no articles devoted to the problem of reviewing spelling systems for the Russian language at the time of this writing. In this regard, an urgent task is to analyze the spelling systems for the Russian language among existing solutions

Materials and methods: in the article methods are used: analysis of hierarchies, Bord's method, method of estimation by the indicator of error of recognition of correspondences; The software libraries of the following spelling systems were used: Yandex.Speller, Google Spellchecker, Hunspell, LanguageTool; dictionary composed of authors with a size of more than 200,000 was used; Python programming language was also used for the software implementation of the estimation methodology.

Results: a technique for estimating automatic spelling systems of the Russian language; a description of the selected systems is given; from the set of solutions the best ones are chosen; a conclusion is drawn about their quality on the basis of optimal criteria.

Conclusion: a technique for estimating automatic spelling systems was proposed. Using this technique, we can conclude that the best solutions are Hunspell and LanguageTool.

The obtained results create the necessary prerequisites for the development of an automatic spelling system for the Russian language.

РЕЗЮМЕ

Я. С. Пикалёв, А. С. Вовнянко, И. Я. Денищенко

Анализ автоматических систем проверки правописания русского языка

История вопроса, исходные данные: разработка систем проверки правописания, показывающей оптимальный результат, является одной из ключевых проблем направления обработки естественного языка (natural language processing, NLP). Актуальность данной проблемы подтверждается широким внедрением систем проверки правописания в большинство видов программных продуктов (текстовые редакторы, системы автоматического перевода и т.п.), а также использованием исследователями и разработчиками для других задач NLP и искусственного интеллекта. Статьи, посвященные задаче обзора систем проверки правописания для русского языка, на момент написания данной работы отсутствуют. В связи с этим актуальной задачей является анализ систем проверки правописания для русского языка среди существующих решений.

Материалы и методы: в статье использованы методы: анализа иерархий, метод Борда, метод оценки по показателю ошибки распознавания соответствий; были использованы программные библиотеки следующих систем проверки правописания: Яндекс. Спеллер, Google Spellchecker, Hunspell, LanguageTool; был использован авторский словарь, содержащий более 200000 слов; также был использован язык программирования Python для программной реализации методики оценивания.

Результаты: предложена методика оценивания автоматических систем проверки правописания русского языка; дано описание отобранным системам; из множества решений выбраны наилучшие; сделан вывод об их качестве на основании оптимальных критериев.

Заключение: была предложена методика оценивания автоматических систем правописания. Используя данную методику, можно сделать вывод, что наилучшими решениями являются Hunspell и LanguageTool.

Полученные результаты создают необходимые предпосылки для разработки автоматической системы правописания русского языка.

Статья поступила в редакцию 20.04.2018.