Исправление поисковых запросов

Ильвохин Дмитрий

25 марта 2019 г.

Запросы с ошибками

Около 11% поисковых запросов содержат ошибки.

Примеры ошибок в одном из самых популярных запросов рунета «вконтакте»:

- drjynfrnt (154),
- вкантакте (53),
- вконтак (34),
- вконтатке (22),
- в контак (18),
- вконта (14),
- вконтакти (13),
- вконтаке (13),
- в конта (11),
- вконакте (10).

Результаты поиска при ошибке в запросе

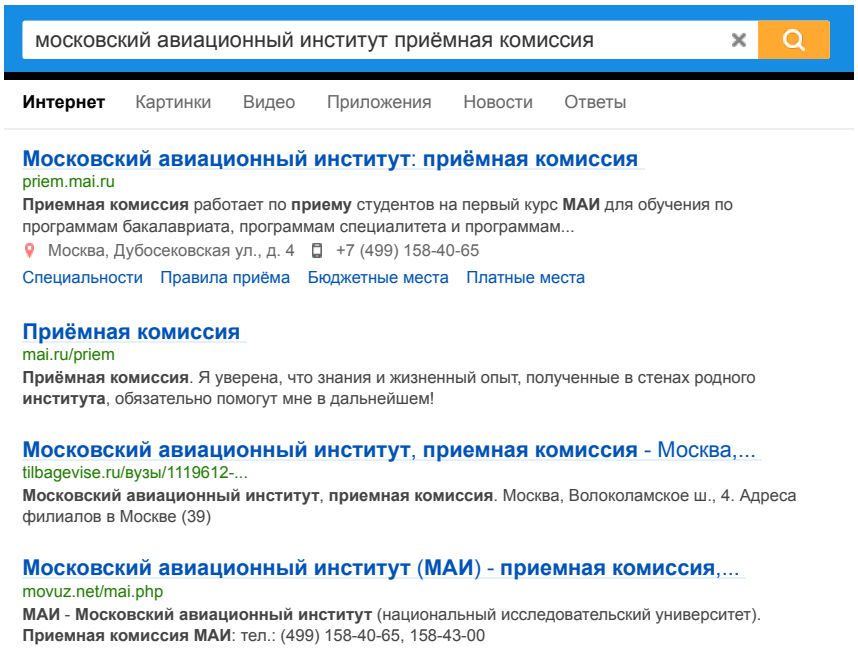


Рис. 1: Результаты по запросу «московский авиационный институт приёмная комиссия»

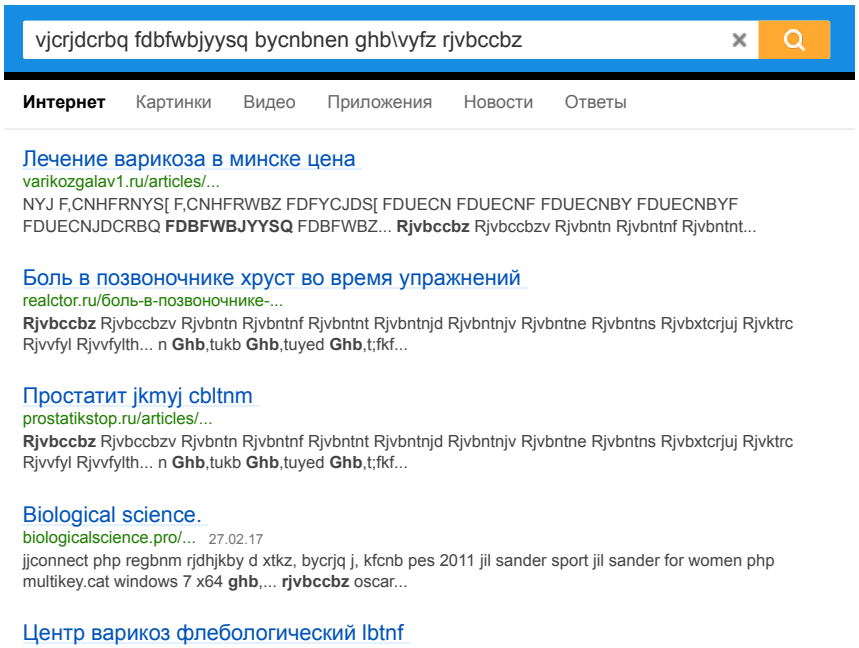


Рис. 2: Результаты по тому же запросу в неверной раскладке клавиатуры

Типы ошибок

Замена и пропуск букв

Замена

- «вкантакте» → «вконтакте»,
- «коталог орифлейм» → «каталог орифлейм»,

Пропуск

- «одноклаcники» → одноклассники,
- «скчать игру» → «скачать игру».

Типы ошибок

Вставка и перестановка букв

Вставка

- «таныцы айренби видео» → «танцы айренби видео».

Перестановка

- «скачтаь медиагет» → «скчатаь медиагет»,
- «купить барслет» → «купить браслет».

Типы ошибок

Разбиение на слова

Склейка слов

- «голос 5 сезон 6выпуск» → «голос 5 сезон 6 выпуск»,
- «неработаетпробел» → «не работает пробел».

Расклейка слов

- «мак бук и вирусы» → «макбук и вирусы»,
- «ю туб» → «ютуб».

Типы ошибок

Раскладка клавиатуры и транслит

Неверная раскладка клавиатуры

- «jlyjrkfccybrb» → «одноклассники»,
- «мл» → «vk».

Транслитерация

- «kupit televizor» → «купить телевизор»,
- «мейл ру» → «mail ru».

Типы ошибок: сложные случаи

Неправильная раскладка клавиатуры?

$$\text{vnc} \longrightarrow \text{мтс}$$

Типы ошибок: сложные случаи

Неправильная раскладка клавиатуры?

$$\text{vnc} \longrightarrow \text{мтс}$$

VNC — Virtual Network Computing, система удалённого доступа к рабочему столу компьютера.

Типы ошибок: сложные случаи

Опечатка?

андекс $\longrightarrow$ яндекс

Типы ошибок: сложные случаи

Опечатка?

андекс $\longrightarrow$ яндекс

Ándекс — посёлок в Германии, который известен благодаря находящемуся там бенедиктинскому монастырю.

Типы ошибок: сложные случаи

Явная ошибка?

п<u>а</u>года $\longrightarrow$ п<u>о</u>года

Типы ошибок: сложные случаи

Явная ошибка?

пагода $\longrightarrow$ погода

Па́года — буддийское или индуистское сооружение культового характера.

Типы ошибок: сложные случаи

Явная ошибка?

Рис. 3: Большая пагода в Лондоне

Виды исправлений

Автоисправление

Автоисправление — надежное исправление, в запросе точно допущена ошибка, будут показаны результаты по исправленному запросу.

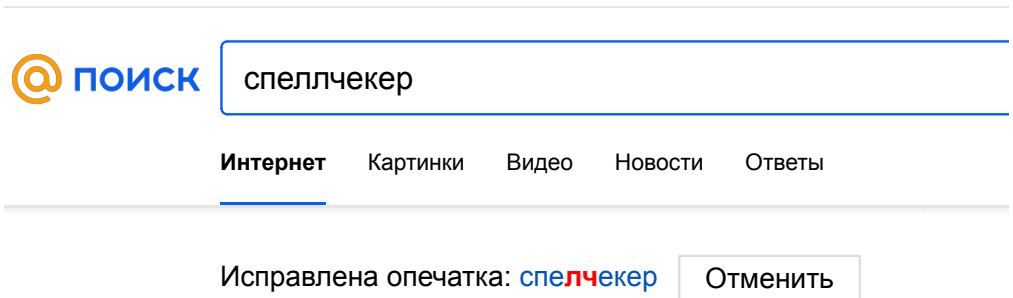


Рис. 4: Пример автоисправления: оригинальный запрос: «спелчекер», автоисправление: «спеллчекер»

Виды исправлений

Подсказка

Подсказка — ненадежное исправление, возможно, в запросе допущена ошибка, будут показаны результаты по оригинальному запросу и предложение исправить запрос.

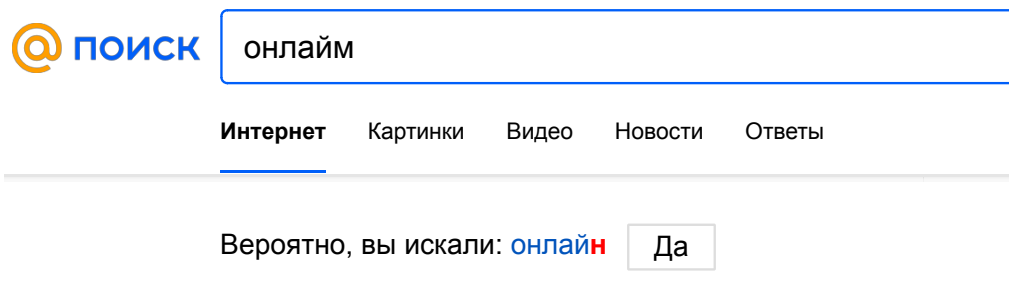


Рис. 5: Пример подсказки, оригинальный запрос: «онлайм», подсказка: «онлайн»

Виды исправлений

Смешение выдач

Смешение — скорее всего в запросе допущена ошибка, будут показаны результаты как по оригинальному запросу, так и по исправленному.

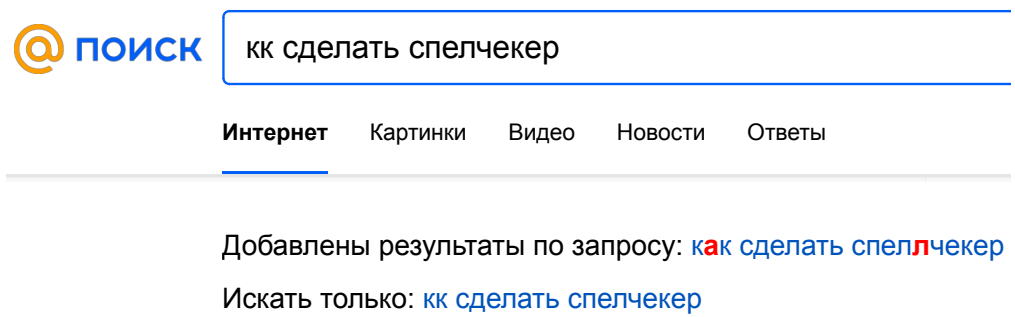


Рис. 6: Пример смешения, оригинальный запрос: «кк сделать спелчекер», исправление: «как сделать спеллчекер»

Модель зашумленного канала[1]

Задача

- Σ — алфавит;
- Σ^* — множество конечных строк над Σ;
- $D \subseteq \Sigma^*$ — словарь корректных слов;
- $q \notin D$, $c \in D$ — строки над Σ.

Найти

$$\arg\max_{c \in D} P(c|q).$$

[1] Brill E., Moore R. C. An improved error model for noisy channel spelling correction, 2000.

Модель зашумленного канала

$$\arg\max_{c \in D} P(c|q) = \arg\max_{c \in D} \frac{P(q|c)P(c)}{P(q)} = \arg\max_{c \in D} P(q|c)P(c).$$

Интерпретация

- $P(c)$ — модель источника (модель языка),
- $P(q|c)$ — модель канала (модель ошибок).

Спеллчекер Питера Норвига (Peter Norvig)[2]

- Словарь: слова из книг проекта Gutenberg, самые частотные слова из Wiktionary и British National Corpus.
- Модель языка на основе частот слов в корпусе.
- Модель ошибок на основе редакционного расстояния.

[2] https://norvig.com/spell-correct.html, 2007.

Спеллчекер Питера Норвига

Основная идея

```
def P(word, N=sum(WORDS.values())):
    return WORDS[word] / N

def correction(word):
    return max(candidates(word), key=P)

def candidates(word):
    return (known([word]) or
            known(edits1(word)) or
            known(edits2(word)) or [word])

def known(words):
    return set(w for w in words if w in WORDS)
```

Рис. 7: «Сердце» спеллчекера

Общая схема работы

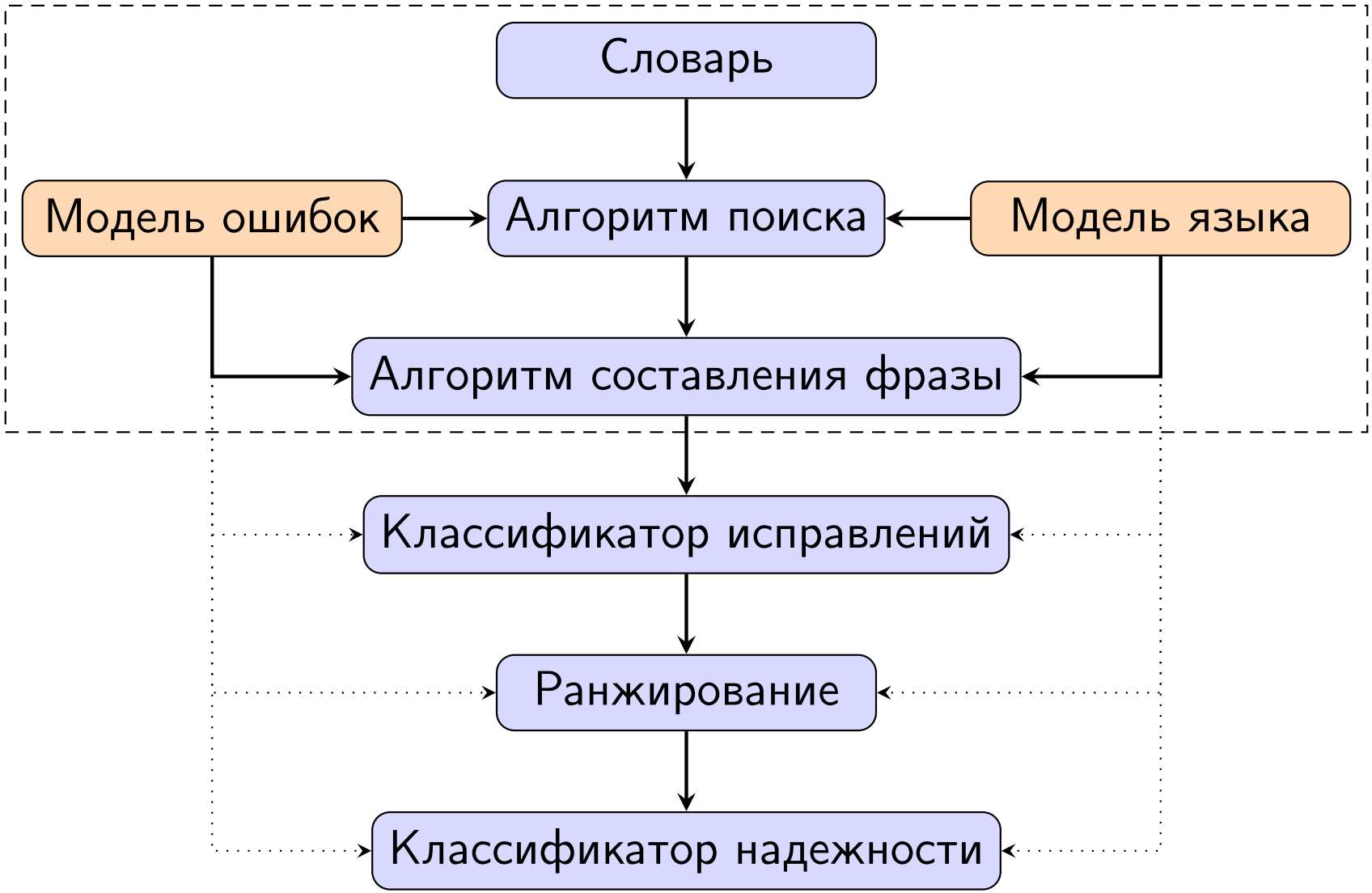


Рис. 8: Составные части системы и их связь

Модель ошибок[3]

Идея

$P(q|c) \propto EditDistance(q, c)$

- Расстояние Левенштейна.
- Расстояние Дамерау–Левенштейна (транспозиция — одна операция).
- Взвесить операции редактирования.
- Вес зависит от контекста и самой операции («тся» $\rightarrow$ «ться»).

[3] Ahmad F., Kondrak G. Learning a spelling error model from search query logs, 2005.

Модель ошибок

Сбор данных

Нужны пары *запрос* $\rightarrow$ *исправление*.

Использование асессоров

- Выдать запрос, попросить исправить.
- Выдать запрос и исправление, попросить отобрать хорошие.
- Выдать корректный тест, попросить перепечатать.
- Выдать аудиофайл, попросить набрать.

Модель ошибок

Автоматический сбор данных

Нужны пары *запрос* $\rightarrow$ *исправление*.

- Кликовые данные.
  - Отказы и согласия с исправлениями на страницах с результатами.
  - Интерливинг.
- Искусственная генерация данных.

Модель ошибок

Построение

Редакционное расстояние между двумя строками a и b.

$$\mathrm{lev}_{a,b}(i,j) = \begin{cases} \max(i,j) & \text{if } \min(i,j) = 0, \\ \min \begin{cases} \mathrm{lev}_{a,b}(i-1,j) + 1 \\ \mathrm{lev}_{a,b}(i,j-1) + 1 \\ \mathrm{lev}_{a,b}(i-1,j-1) + 1_{(a_i \neq b_j)} \end{cases} & \text{otherwise.} \end{cases}$$

- Используем алгоритм Вагнера–Фишера (Wagner–Fischer).
- По матрице *lev* получаем редакционное предписание.
- По редакционному предписанию для всех пар сохраняем статистику.

Модель ошибок

Построение, модели разных уровней

- Нулевого уровень: вероятность операции данного типа (вставка, замена, удаление)?
- Первый уровень: нулевой уровень и дополнительно учитывается текущая буква.
- Второй уровень: первый уровень и дополнительно учитывается предыдущая буква.
- ...

Модель ошибок

Построение, пример

«президент» $\rightarrow$ «перзидеед»

Таблица 1: Построение модели ошибок

M	M	D	M	I	M	M	M	M	R	R	M
^	п	р	е	␣	з	и	д	е	н	т	$
^	п	␣	е	р	з	и	д	е	е	д	$

- «^п» $\rightarrow$ «^п» — 1,
- «пр» $\rightarrow$ «п␣» — 1,
- ...
- «т$» $\rightarrow$ «д$» — 1.

Модель ошибок

Смешение моделей разного уровня

$$\alpha Z + \beta F + \gamma S,$$

где Z, F, S — модели нулевого, первого и второго уровней, $\alpha, \beta, \gamma \in [0, 1], \alpha + \beta + \gamma = 1$.

Модель языка

Пусть $Q = w_1, w_2, \ldots, w_n$ — запрос, состоящий из n слов,

$$P(Q) = P(w_1)P(w_2|w_1)P(w_3|w_1 w_2)\ldots P(w_n|w_1 \ldots w_{n-1}).$$

Униграмная

$$P(Q) \approx P(w_1)P(w_2)P(w_3)\ldots P(w_n).$$

Биграмная

$$P(Q) \approx P(w_1)P(w_2|w_1)P(w_3|w_2)\ldots P(w_n|w_{n-1}).$$

Техники сглаживания[4]

Проблема

Пусть $Q = w_1, w_2, \ldots, w_n$ — запрос, состоящий из n слов, если $\exists w_k \notin D \Rightarrow P(Q) = 0$.

[4] Chen S. F., Goodman J. An empirical study of smoothing techniques for language modeling, 1999.

Источники данных

- Словари, книги, тексты новостей.
- Логи запросов.
- Слова из веб-документов.

Алгоритм поиска

Префиксное дерево

Идея

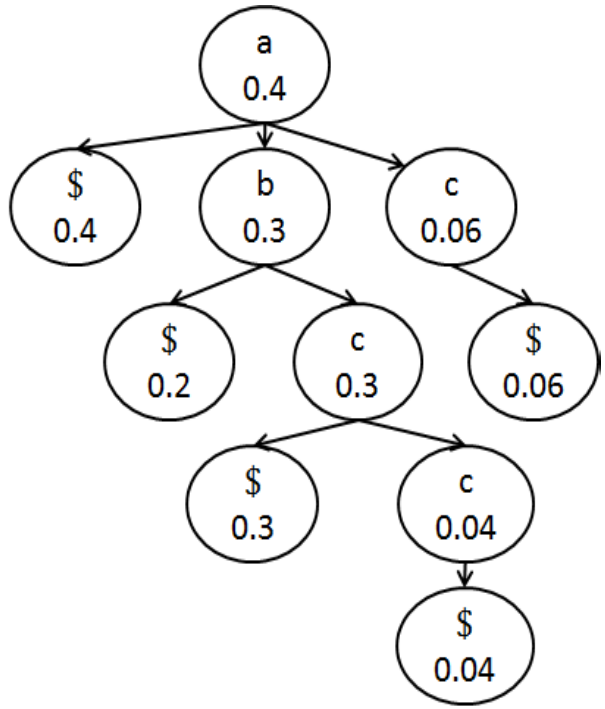
- Построим префиксное дерево (бор, trie) для слов из словаря.
- Запустим на нем A^*.
- Придумаем эвристик, чтобы быстро работало.

Префиксное дерево

Эвристики[5]

Query	Count
a	20
ab	10
ac	3
abc	15
abcc	2

a



b

Рис. 9: (a) — набор строк с частотами, (b) — префиксное дерево над этим набором

[5] Duan H., Hsu B. J. P. Online spelling correction for query completion, 2011.

Префиксное дерево

Поиск[6]

```
Input:  Query trie T, transformation model Θ, integer k, query prefix q̄
Output: Top k completion suggestions of q̄
A   List l = new List()
B   PriorityQueue pq = new PriorityQueue()
C   pq.Enqueue(new Path(0, T.Root, [], 1))
D   while (!pq.Empty())
E       Path π = pq.Dequeue()
F       if (π.Pos < |q̄|)                          // Transform input query
G           foreach (Transfeme t in GetTransformations(π, q̄, T, Θ))
H               int i      = π.Pos + t.Output.Length
I               Node n     = π.Node.FindDescendant(t.Input)
J               History h  = π.Hist + t
K               Prob p     = π.Prob × (n.Prob / π.Node.Prob) ×
                             P(t, π.Hist; Θ)
L               pq.Enqueue(new Path(i, n, h, p))
M       else                                      // Extend input query
N           if (π.Node.IsLeaf())
O               l.Add(π.Node.Query)
P               if (l.Count ≥ k)
Q                   return l
R           else
S               foreach (Transfeme t in GetExtensions(π, T, Θ))
T                   int i      = π.Pos + t.Output.Length
U                   Node n     = π.Node.FindDescendant(t.Input)
V                   History h  = π.Hist + t
W                   Prob p     = π.Prob × (n.Prob / π.Node.Prob)
X                   pq.Enqueue(new Path(i, n, h, p))
Y   return l
```

Рис. 10: Псевдокод алгоритма поиска

[6] Duan H., Hsu B. J. P. Online spelling correction for query completion, 2011.

Оптимизации

Идеи для ускорения

- Используем только K-лучших переходов для каждой позиции в запросе.
- Делаем переходы в узлы, если вероятность в них выше определенного процента максимальной вероятности в позиции.
- Ограничение количества операций редактирования — функция от длины запроса.
- Предподсчет ответов для частых запросов.

Алгоритм поиска

Альтернативный подход, фонетические алгоритмы

Metaphone — фонетический алгоритм для индексирования слов по их звучанию с учётом основных правил английского произношения.

Фонетические алгоритмы

Вариант для русского языка[7]

Таблица 2: Аналог Metaphone для русского языка

Код	Буквы
1	а, о, ы, у, я
3	и, е, ё, ю, я, э
5	б, п
6	в, ф
7	д, т

Код	Буквы
8	г, к, х
9	л
10	р
11	м
12	н

Код	Буквы
13	з, с
14	й
15	щ, ч
16	ж, ш
17	ц

[7] Sorokin A. A., Shavrina T. O. Automatic spelling correction for Russian social media texts, 2016.

Фонетические алгоритмы

Пример работы

Замены по таблице и несколько других несложных правил.

Пример

- «скачать» $\rightarrow \{13, 8, 1, 15, 1, 7\}$,
- «скочять» $\rightarrow \{13, 8, 1, 15, 1, 7\}$.

Алгоритм составления фразы

Идея

Пусть исходный запрос Q был разбит на слова w_1, w_2, ..., w_n. Для каждого слова были найдены вероятные кандидаты:

- $w_1 : c_{1,1}, c_{1,2}, \ldots, c_{1,k}$,
- $w_2 : c_{2,1}, c_{2,2}, \ldots, c_{2,k}$,
- ...
- $w_n : c_{n,1}, c_{n,2}, \ldots, c_{n,k}$.

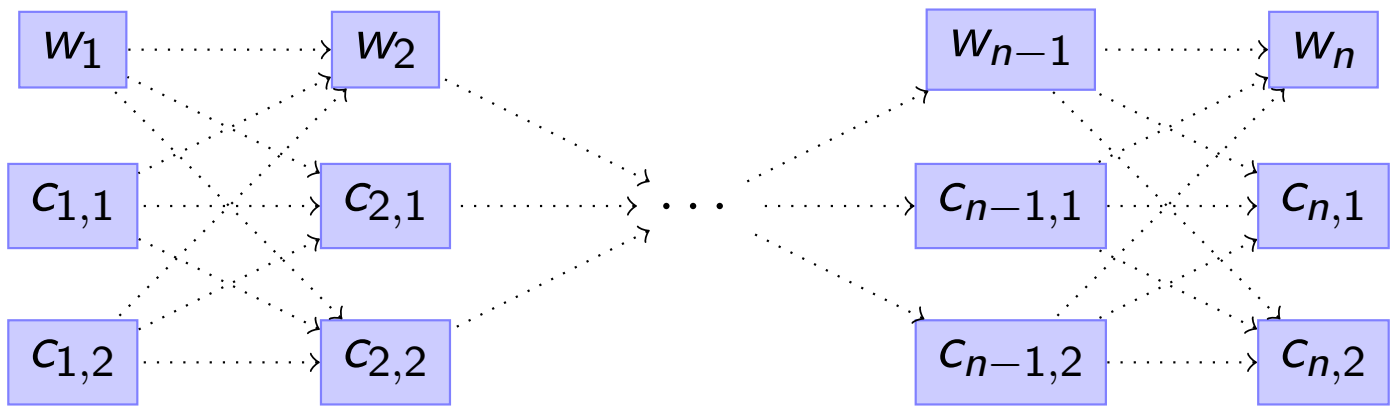


Рис. 11: Схема формирования фразы

Алгоритм составления фразы

Имеем

- Скрытую Марковскую модель и последовательность наблюдений.
- Веса переходов — вероятности языковой модели $P(w_i|w_{i-1})$.

Найти

Наиболее вероятную последовательность скрытых состояний.

Алгоритм составления фразы

Алгоритм Витерби

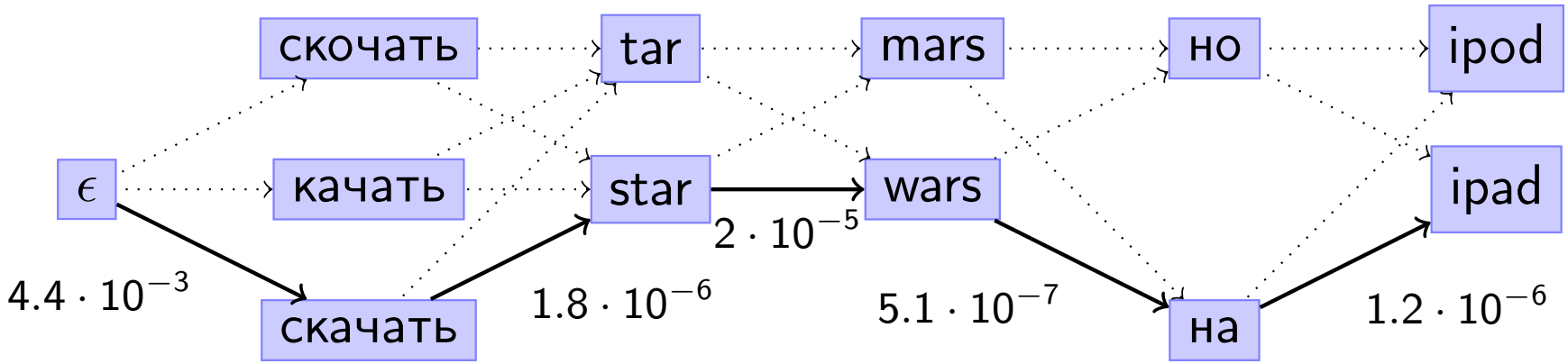


Рис. 12: Пример работы алгоритма

Биграмная языковая модель

$$viterbi_{i,j} = \max_{0 \leq k < |C_{i-1}|} viterbi_{i-1,k} \times P(c_{i,j}|c_{i-1,k}).$$

Другие типы исправлений

Общая схема работы похожа на исправление слов.

Разбиение слова

- Операции: добавить, убрать разделитель.
- Оцениваем результат с помощью модели языка.

Неверная раскладка

- Собственное разбиение на слова («rf ̲fxjr» $\rightarrow$ |ка|б|ачок|).
- Дополнительная операция редактирования: смена раскладки (зависит от типа клавиатуры).

Классификатор исправлений

- «нечеткий пиоск строки» $\rightarrow$ «нечеткий поиск строки»,
- «нечеткий пиоск строки» $\not\rightarrow$ «нечеткий писк строки»,
- «нечеткий пиоск строки» $\not\rightarrow$ «нечеткий киоск строки».

Признаки

- Пословные.
- Позапросные.
- По типам исправлений (исправление в имени, изменение формы слова).

Ранжирование

Сложности

- Нужно уметь ранжировать разнородные исправления.
- Сбор обучающих данных.

Итеративные исправления

Более 90% ошибок исправляются за одну итерацию.

1. «методы государственное подерки литиратура»,
2. «методы государственное подерки литературы»,
3. «методы государственной поддержки литературы».

Надежность исправлений

Решает нужно ли автоматическое исправление или подсказка.

Точно нужно исправлять, но во что?

- «поск» $\rightarrow$ {«поиск», «писк»},
- «атташа» $\rightarrow$ {«наташа», «атташе}.

Вероятность исправления большая, но не факт, что есть ошибка.

- «vnc» $\rightarrow$ «мтс»,
- «пагода» $\rightarrow$ «погода».

Домашнее задание

Реализовать систему для исправления **слов**.

Kaggle-соревнование

https://www.kaggle.com/c/itmo-spelling-correction-2019

Метрика

Среднее расстояние Левенштейна.

Домашнее задание

Ограничения

- Максимальное количество баллов — 25.
- Решения хуже «No Fixes At All» получают 0 баллов.
- Не более шести посылок в сутки.
- Чужие решения сдавать нельзя.
- Deadline: 04/15/2019 11:59 PM UTC.

Домашнее задание

Баллы

Код решения, его короткое описание и kaggle-хэндл нужно прислать на электронную почту.

Распределение баллов

- Префиксное дерево и эвристики с лекции — 10 баллов.
- Фонетические алгоритмы — 3 балла.
- Модель языка — 2 балла.
- Модель ошибок — 5 баллов (все уровни).
- Клёвые эвристики — 5 баллов (не менее двух).

Домашнее задание

Контакты

- **Электронная почта: d.ilvokhin@corp.mail.ru**
- Telegram: https://t.me/r3tsky