Математические модели в лингвистике

Исправление опечаток

Мати Пентус, Александр Пиперски, Алексей Сорокин

МГУ им. М. В. Ломоносова, межфакультетский курс,
осенний семестр 2015–2016 учебного года

Определение задачи

Исправление опечаток

Исправление опечаток (в широком смысле) — выявление ошибок правописания и поиск возможных вариантов их исправления.

Определение задачи

Исправление опечаток

Исправление опечаток (в широком смысле) — выявление ошибок правописания и поиск возможных вариантов их исправления.

- Опечатки и орфографические ошибки содержатся в любом достаточно длинном тексте.

Определение задачи

Исправление опечаток

Исправление опечаток (в широком смысле) — выявление ошибок правописания и поиск возможных вариантов их исправления.

- Опечатки и орфографические ошибки содержатся в любом достаточно длинном тексте.

Решено было не допустить ни одной ошибки. Держали двадцать корректур. И все равно на титульном листе было напечатано: Британская энциклопудия.

Илья Ильф, «Записные книжки»

Определение задачи

Исправление опечаток

Исправление опечаток (в широком смысле) — выявление ошибок правописания и поиск возможных вариантов их исправления.

- Опечатки и орфографические ошибки содержатся в любом достаточно длинном тексте.

Решено было не допустить ни одной ошибки. Держали двадцать корректур. И все равно на титульном листе было напечатано: Британская энциклопудия.

Илья Ильф, «Записные книжки»

- По запросу “сваего мнения” google.ru находит 689 результатов.

Применения исправления опечаток

- Информационный поиск (коррекция запросов)

Применения исправления опечаток

- Информационный поиск (коррекция запросов)
- Программы проверки правописания в текстовых редакторах

Применения исправления опечаток

- Информационный поиск (коррекция запросов)
- Программы проверки правописания в текстовых редакторах
- Нормализация текста (коррекция стилистических вариантов, актуальна для социальных медиа)

Применения исправления опечаток

- Информационный поиск (коррекция запросов)
- Программы проверки правописания в текстовых редакторах
- Нормализация текста (коррекция стилистических вариантов, актуальна для социальных медиа)

Применения исправления опечаток

- Информационный поиск (коррекция запросов)
- Программы проверки правописания в текстовых редакторах
- Нормализация текста (коррекция стилистических вариантов, актуальна для социальных медиа)
- Распознавание звучащей речи, изображений...

Применения исправления опечаток

- Информационный поиск (коррекция запросов)
- Программы проверки правописания в текстовых редакторах
- Нормализация текста (коррекция стилистических вариантов, актуальна для социальных медиа)
- Распознавание звучащей речи, изображений...
- Компаративистика («опечатки» — эволюционные изменения в словах языка или различия между родственными языками)

Причины опечаток

- Разновидности опечаток (в широком смысле):
  - Орфографические ошибки

Причины опечаток

- Разновидности опечаток (в широком смысле):
  - Орфографические ошибки
  - Типографские ошибки (опечатки в узком смысле/описки)

Причины опечаток

- Разновидности опечаток (в широком смысле):
  - Орфографические ошибки
  - Типографские ошибки (опечатки в узком смысле/описки)
  - Когнитивные ошибки
    (смешение понятий, “предать” $\leftrightarrow$ “придать”)

Причины опечаток

- Разновидности опечаток (в широком смысле):
  - Орфографические ошибки
  - Типографские ошибки (опечатки в узком смысле/описки)
  - Когнитивные ошибки
    (смешение понятий, “предать” $\leftrightarrow$ “придать”)
  - Ошибки при записи речи “на слух”.

Причины опечаток

- Разновидности опечаток (в широком смысле):
  - Орфографические ошибки
  - Типографские ошибки (опечатки в узком смысле/описки)
  - Когнитивные ошибки
    (смешение понятий, “предать” $\leftrightarrow$ “придать”)
  - Ошибки при записи речи “на слух”.
  - Транслитерационные ошибки (в иноязычных словах/именах собственных).

Причины опечаток

- Разновидности опечаток (в широком смысле):
  - Орфографические ошибки
  - Типографские ошибки (опечатки в узком смысле/описки)
  - Когнитивные ошибки
    (смешение понятий, “предать” $\leftrightarrow$ “придать”)
  - Ошибки при записи речи “на слух”.
  - Транслитерационные ошибки (в иноязычных словах/именах собственных).
  - Региональная/стилистическая вариативность
    (“colour” $\leftrightarrow$ “color”, “dialogue” $\leftrightarrow$ “dialog”).

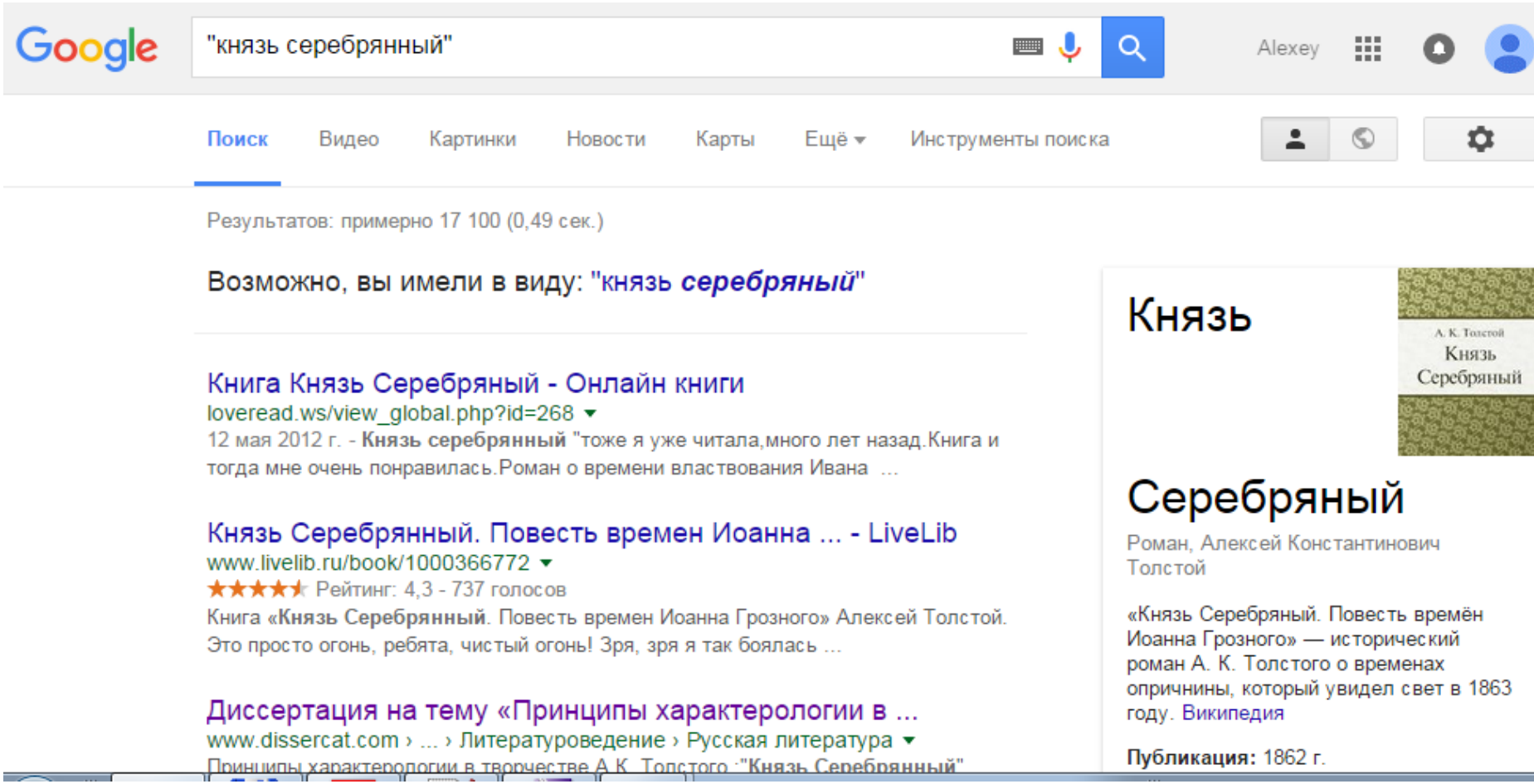
Причины опечаток

- Разновидности опечаток (в широком смысле):
  - Орфографические ошибки
  - Типографские ошибки (опечатки в узком смысле/описки)
  - Когнитивные ошибки
    (смешение понятий, “предать” $\leftrightarrow$ “придать”)
  - Ошибки при записи речи “на слух”.
  - Транслитерационные ошибки (в иноязычных словах/именах собственных).
  - Региональная/стилистическая вариативность
    (“colour” $\leftrightarrow$ “color”, “dialogue” $\leftrightarrow$ “dialog”).
- Чаще всего ошибки локальны (затрагивают один-два символа)

Причины опечаток

- Разновидности опечаток (в широком смысле):
  - Орфографические ошибки
  - Типографские ошибки (опечатки в узком смысле/описки)
  - Когнитивные ошибки (смешение понятий, “предать” $\leftrightarrow$ “придать”)
  - Ошибки при записи речи “на слух”.
  - Транслитерационные ошибки (в иноязычных словах/именах собственных).
  - Региональная/стилистическая вариативность (“colour” $\leftrightarrow$ “color”, “dialogue” $\leftrightarrow$ “dialog”).
- Чаще всего ошибки локальны (затрагивают один-два символа)
- Однако может влиять и более широкий контекст (“ться” $\rightarrow$ *“цца”, “ян” $\rightarrow$ “янн/енн” в суффиксе прилагательного)

Пример


Google
"князь серебрянный"
Alexey
Поиск
Видео
Картинки
Новости
Карты
Ещё
Инструменты поиска
Результатов: примерно 17 100 (0,49 сек.)
Возможно, вы имели в виду: "князь серебряный"
Книга Князь Серебряный - Онлайн книги
loveread.ws/view_global.php?id=268
12 мая 2012 г. - Князь серебрянный "тоже я уже читала,много лет назад.Книга и тогда мне очень понравилась.Роман о времени властвования Ивана ...
Князь Серебрянный. Повесть времен Иоанна ... - LiveLib
www.livelib.ru/book/1000366772
Рейтинг: 4,3 - 737 голосов
Книга «Князь Серебрянный. Повесть времен Иоанна Грозного» Алексей Толстой. Это просто огонь, ребята, чистый огонь! Зря, зря я так боялась ...
Диссертация на тему «Принципы характерологии в ...
www.dissercat.com › ... › Литературоведение › Русская литература
Принципы характерологии в творчестве А.К. Толстого :"Князь Серебрянный"
Князь
А. К. Толстой
Князь Серебряный
Серебряный
Роман, Алексей Константинович Толстой
«Князь Серебряный. Повесть времён Иоанна Грозного» — исторический роман А. К. Толстого о временах опричнины, который увидел свет в 1863 году. Википедия
Публикация: 1862 г.

Методы исправления опечаток

- Как исправлять опечатки?

Методы исправления опечаток

- Как исправлять опечатки?
- Можно искать близкие слова в словаре.

Методы исправления опечаток

- Как исправлять опечатки?
- Можно искать близкие слова в словаре.
- Что значит “близкие”?
- Требуется задать функцию расстояния на множестве слов.

Методы исправления опечаток

- Как исправлять опечатки?
- Можно искать близкие слова в словаре.
- Что значит “близкие”?
- Требуется задать функцию расстояния на множестве слов.
- Кроме того хочется, чтобы эту функцию было легко вычислять или хотя бы оценивать.

Модель близости слов

Расстояние Левенштейна

Расстояние Левенштейна $\rho_L(u, v)$ между словами u и v — минимальное число замен, вставок и удалений, необходимых, чтобы получить v из u.

Модель близости слов

Расстояние Левенштейна

Расстояние Левенштейна $\rho_L(u, v)$ между словами u и v — минимальное число замен, вставок и удалений, необходимых, чтобы получить v из u.

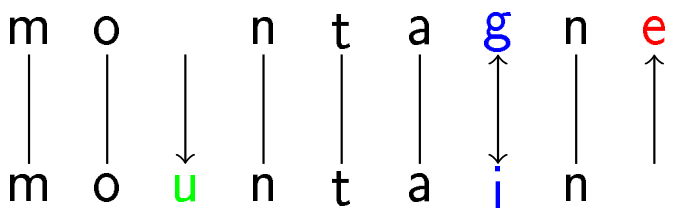
Появилась в работах В. И. Левенштейна (1965) и F. Damerau (1964).

Модель близости слов

Расстояние Левенштейна

Расстояние Левенштейна $\rho_L(u, v)$ между словами u и v — минимальное число замен, вставок и удалений, необходимых, чтобы получить v из u.

Появилась в работах В. И. Левенштейна (1965) и F. Damerau (1964).

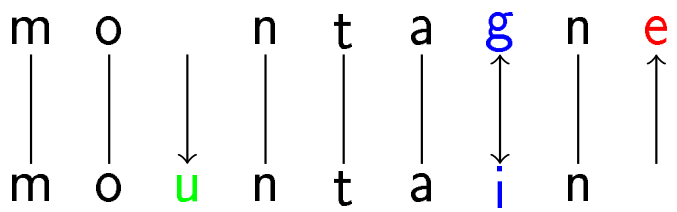


Модель близости слов

Расстояние Левенштейна

Расстояние Левенштейна $\rho_L(u, v)$ между словами u и v — минимальное число замен, вставок и удалений, необходимых, чтобы получить v из u.

Появилась в работах В. И. Левенштейна (1965) и F. Damerau (1964).



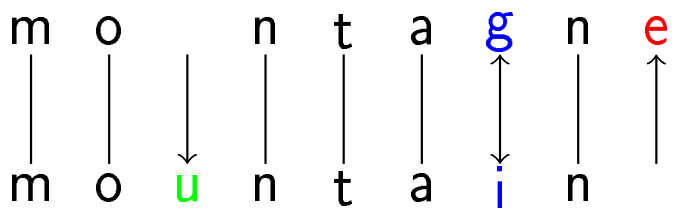
$$d(montagne, mountain) = 3$$

Модель близости слов

Расстояние Левенштейна

Расстояние Левенштейна $\rho_L(u, v)$ между словами u и v — минимальное число замен, вставок и удалений, необходимых, чтобы получить v из u.

Появилась в работах В. И. Левенштейна (1965) и F. Damerau (1964).



$$d(montagne, mountain) = 3$$

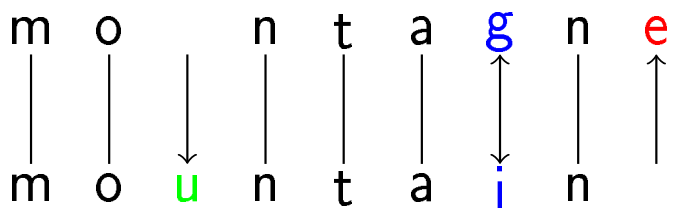
- Можно добавить перестановку соседних символов (с некоторыми ограничениями).

Модель близости слов

Расстояние Левенштейна

Расстояние Левенштейна $\rho_L(u, v)$ между словами u и v — минимальное число замен, вставок и удалений, необходимых, чтобы получить v из u.

Появилась в работах В. И. Левенштейна (1965) и F. Damerau (1964).



$$d(montagne, mountain) = 3$$

- Можно добавить перестановку соседних символов (с некоторыми ограничениями).
- Можно по-разному “штрафовать” за разные изменения.

Вычисление расстояния Левенштейна

Обозначения:

- $w = w_0 \dots w_{n-1}$ — слово, $|w| = n$ — длина слова.

Вычисление расстояния Левенштейна

Обозначения:

- $w = w_0 \dots w_{n-1}$ — слово, $|w| = n$ — длина слова.
- $w[i]$ — i-ый символ слова, $w[i, j]$ — подслово с i-ой по j-ую позицию (не включая j).

Вычисление расстояния Левенштейна

Обозначения:

- $w = w_0 \dots w_{n-1}$ — слово, $|w| = n$ — длина слова.
- $w[i]$ — i-ый символ слова, $w[i, j]$ — подслово с i-ой по j-ую позицию (не включая j).
- $w[, j]$ — префикс по j-ую позицию (не включая j).
- $w[i,]$ — суффикс с i-ой позиции (включая i).

Вычисление расстояния Левенштейна

Обозначения:

- $w = w_0 \dots w_{n-1}$ — слово, $|w| = n$ — длина слова.
- $w[i]$ — i-ый символ слова, $w[i, j]$ — подслово с i-ой по j-ую позицию (не включая j).
- $w[, j]$ — префикс по j-ую позицию (не включая j).
- $w[i,]$ — суффикс с i-ой позиции (включая i).

Идея алгоритма: будем вычислять $d_{ij} = \rho(u[, i], v[, j])$ рекурсивно через значения для меньших i, j. Если $|u| = m, |v| = n$, то ответом будет d_{mn}.

Рекурсивная формула

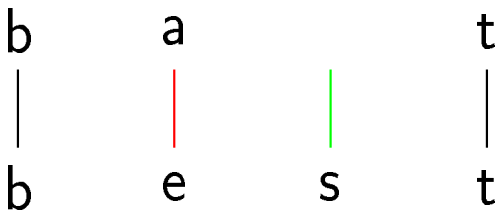
$$
\begin{aligned}
\rho(u[,i],v[,0]) &= i,\\
\rho(u[,0],v[,j]) &= j,\\
\rho(u[,i],v[,j]) &= \min\big(\rho(u[,i-1],v[,j-1]) + [\![u[i-1] \neq v[j-1]]\!],\\
&\qquad \rho(u[,i],v[,j-1]) + 1,\\
&\qquad \rho(u[,i-1],v[,j]) + 1\big)
\end{aligned}
$$

Рекурсивная формула

$$\begin{aligned}
\rho(u[,i],v[,0]) &= i,\\
\rho(u[,0],v[,j]) &= j,\\
\rho(u[,i],v[,j]) &= \min\big(\rho(u[,i-1],v[,j-1]) + [\![u[i-1] \neq v[j-1]]\!],\\
&\qquad \rho(u[,i],v[,j-1]) + 1,\\
&\qquad \rho(u[,i-1],v[,j]) + 1\big)
\end{aligned}$$

Возможные случаи:

- $\rho(u[,i],v[,j]) = \rho(u[,i-1],v[,j-1])$:
  $d(bat, best) = d(ba, bes) = 2$.

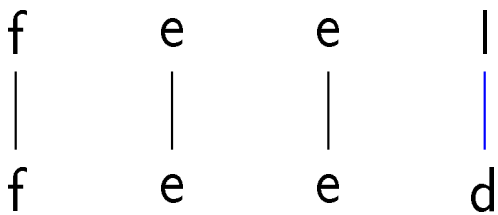


Рекурсивная формула

$$\begin{aligned}
\rho(u[,i],v[,0]) &= i,\\
\rho(u[,0],v[,j]) &= j,\\
\rho(u[,i],v[,j]) &= \min\big(\rho(u[,i-1],v[,j-1]) + [\![u[i-1] \neq v[j-1]]\!],\\
&\qquad \rho(u[,i],v[,j-1]) + 1,\\
&\qquad \rho(u[,i-1],v[,j]) + 1\big)
\end{aligned}$$

Возможные случаи:

- $\rho(u[,i],v[,j]) = \rho(u[,i-1],v[,j-1])$:
  $d(bat, best) = d(ba, bes) = 2$.
- $\rho(u[,i],v[,j]) = \rho(u[,i-1],v[,j-1]) + 1$:
  $\rho(feel, feed) = \rho(fee, fee) + 1 = 1$.



Рекурсивная формула

$$\begin{aligned}
\rho(u[,i],v[,0]) &= i,\\
\rho(u[,0],v[,j]) &= j,\\
\rho(u[,i],v[,j]) &= \min\big(\rho(u[,i-1],v[,j-1]) + [\![u[i-1] \neq v[j-1]]\!],\\
&\qquad \rho(u[,i],v[,j-1]) + 1,\\
&\qquad \rho(u[,i-1],v[,j]) + 1\big)
\end{aligned}$$

Возможные случаи:

- $\rho(u[,i],v[,j]) = \rho(u[,i-1],v[,j-1])$:
  $d(bat, best) = d(ba, bes) = 2$.
- $\rho(u[,i],v[,j]) = \rho(u[,i-1],v[,j-1]) + 1$:
  $\rho(feel, feed) = \rho(fee, fee) + 1 = 1$.
- $\rho(u[,i],v[,j]) = \rho(u[,i-1],v[,j]) + 1$:
  $\rho(slide, solid) = \rho(slid, solid) + 1 = 2$.

s	∅	l	i	d	e
|	|	|	|	|	|
s	o	l	i	d	∅

Рекурсивная формула

$$
\begin{aligned}
\rho(u[,i],v[,0]) &= i,\\
\rho(u[,0],v[,j]) &= j,\\
\rho(u[,i],v[,j]) &= \min\big(\rho(u[,i-1],v[,j-1]) + [\![u[i-1] \neq v[j-1]]\!],\\
&\qquad \rho(u[,i],v[,j-1]) + 1,\\
&\qquad \rho(u[,i-1],v[,j]) + 1\big)
\end{aligned}
$$

Возможные случаи:

- $\rho(u[,i],v[,j]) = \rho(u[,i-1],v[,j-1])$:
  $d(bat, best) = d(ba, bes) = 2$.
- $\rho(u[,i],v[,j]) = \rho(u[,i-1],v[,j-1]) + 1$:
  $\rho(feel, feed) = \rho(fee, fee) + 1 = 1$.
- $\rho(u[,i],v[,j]) = \rho(u[,i-1],v[,j]) + 1$:
  $\rho(slide, solid) = \rho(slid, solid) + 1 = 2$.
- $\rho(u[,i],v[,j]) = \rho(u[,i],v[,j-1]) + 1$:
  $\rho(site, step) = \rho(site, ste) + 1 = 2$.

Окончательная рекуррентная формула

Лемма

$\rho(ux, vx) = \rho(u, v)$

Окончательная рекуррентная формула

Лемма
$\rho(ux, vx) = \rho(u, v)$

Окончательная формула для расстояния Левенштейна:

$$
\begin{aligned}
\rho(u[,i], v[,0]) &= i, \\
\rho(u[,0], v[,j]) &= j, \\
\rho(u[,i], v[,j]) &= \rho(u[,i-1], v[,j-1]), \text{ если } u[i-1] = v[j-1] \\
\rho(u[,i], v[,j]) &= \min\big(\rho(u[,i-1], v[,j-1]), \\
&\qquad \rho(u[,i], v[,j-1]), \\
&\qquad \rho(u[,i-1], v[,j])\big) + 1, \text{ если } u[i-1] \neq v[j-1]
\end{aligned}
$$

Окончательная рекуррентная формула

Лемма
$\rho(ux, vx) = \rho(u, v)$

Окончательная формула для расстояния Левенштейна:

$$\begin{aligned}
\rho(u[,i], v[,0]) &= i, \\
\rho(u[,0], v[,j]) &= j, \\
\rho(u[,i], v[,j]) &= \rho(u[,i-1], v[,j-1]), \text{ если } u[i-1] = v[j-1] \\
\rho(u[,i], v[,j]) &= \min\big(\rho(u[,i-1], v[,j-1]), \\
&\qquad \rho(u[,i], v[,j-1]), \\
&\qquad \rho(u[,i-1], v[,j])\big) + 1, \text{ если } u[i-1] \neq v[j-1]
\end{aligned}$$

Идея алгоритма: будем заполнять двумерную таблицу D размера $(m+1)\times(n+1)$, где в ячейке с номером (i,j) будет храниться $\rho(u[,i], v[,j])$. Заполняем рекурсивно по возрастанию i и j.

Псевдокод

Вход: Слова u, v длины m, n соответственно.
Выход: $d(u, v)$ — расстояние Левенштейна между u и v.

```
▷ D — таблица размера (m + 1) × (n + 1)
D[0, 0] = 0
for i = 1, ..., m do
    D[i, 0] = i
end for
for j = 1, ..., n do
    D[0, j] = j
end for
for i = 1, ..., m do
    for j = 1, ..., n do
        if u[i − 1] == v[j − 1] then
            D[i, j] = D[i − 1, j − 1] + 1
        else
            d = min(D[i − 1, j − 1], D[i, j − 1], D[i − 1, j])
            D[i, j] = d + 1
        end if
    end for
end for
    return D[m, n]
```

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1						
т	2	2						
о	3	3						
р	4	4						
а	5	5						

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1						
т	2	2						
о	3	3						
р	4	4						
а	5	5						

$$T_{11} = \min(T_{00}, T_{01}, T_{10}) + 1 = 1 \text{ (т.к. с} \neq \text{ш)}.$$

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1					
т	2	2						
о	3	3						
р	4	4						
а	5	5						

$$T_{11} = \min(T_{00}, T_{01}, T_{10}) + 1 = 1 \text{ (т.к. с} \neq \text{ш)}.$$

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1					
т	2	2						
о	3	3						
р	4	4						
а	5	5						

Поскольку “ш” не входит в слово “строка”, получаем:

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1	2	3	4	5	6
т	2	2						
о	3	3						
р	4	4						
а	5	5						

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1	2	3	4	5	6
т	2	2						
о	3	3						
р	4	4						
а	5	5						

$$T_{21} = \min\left(T_{20}, T_{10}, T_{11}\right) + 1 = 2$$

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1	2	3	4	5	6
т	2	2	2					
о	3	3						
р	4	4						
а	5	5						

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1	2	3	4	5	6
т	2	2	2					
о	3	3						
р	4	4						
а	5	5						

$T_{22} = T_{11} = 1$, поскольку $u[1] = v[1] = \text{т}$

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1	2	3	4	5	6
т	2	2	2	1				
о	3	3						
р	4	4						
а	5	5						

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1	2	3	4	5	6
т	2	2	2	1				
о	3	3						
р	4	4						
а	5	5						

Дальше заполняем строку по алгоритму:

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1	2	3	4	5	6
т	2	2	2	1	2	3	4	5
о	3	3						
р	4	4						
а	5	5						

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1	2	3	4	5	6
т	2	2	2	1	2	3	4	5
о	3	3						
р	4	4						
а	5	5						

Заполняем третью строку:

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1	2	3	4	5	6
т	2	2	2	1	2	3	4	5
о	3	3	3	2	2			
р	4	4						
а	5	5						

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1	2	3	4	5	6
т	2	2	2	1	2	3	4	5
о	3	3	3	2	2			
р	4	4						
а	5	5						

Поскольку $u[2] = v[3] = \text{о}$, то $T_{34} = T_{23} = 2$

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1	2	3	4	5	6
т	2	2	2	1	2	3	4	5
о	3	3	3	2	2	2		
р	4	4						
а	5	5						

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1	2	3	4	5	6
т	2	2	2	1	2	3	4	5
о	3	3	3	2	2	2		
р	4	4						
а	5	5						

Действуя по алгоритму, заполняем таблицу до конца:

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1	2	3	4	5	6
т	2	2	2	1	2	3	4	5
о	3	3	3	2	2	2	3	4
р	4	4	4	3	2	3	3	4
а	5	5	5	4	3	3	4	3

Пример вычисления расстояния Левенштейна

$$d(\text{штора}, \text{строка}) = ?$$

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1	2	3	4	5	6
т	2	2	2	1	2	3	4	5
о	3	3	3	2	2	2	3	4
р	4	4	4	3	2	3	3	4
а	5	5	5	4	3	3	4	3

Таким образом, $d(\text{строка}, \text{штора}) = T_{65} = 3$

Восстановление оптимального выравнивания

По таблице можно найти оптимальное выравнивание:

Восстановление оптимального выравнивания

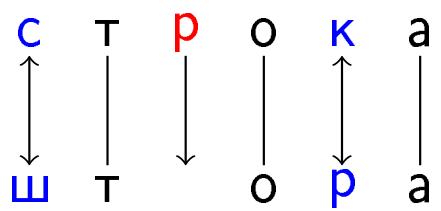
По таблице можно найти оптимальное выравнивание:

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1	2	3	4	5	6
т	2	2	2	1	2	3	4	5
о	3	3	3	2	2	2	3	4
р	4	4	4	3	2	3	3	4
а	5	5	5	4	3	3	4	3

Восстановление оптимального выравнивания

По таблице можно найти оптимальное выравнивание:

			с	т	р	о	к	а
		0	1	2	3	4	5	6
	0	0	1	2	3	4	5	6
ш	1	1	1	2	3	4	5	6
т	2	2	2	1	2	3	4	5
о	3	3	3	2	2	2	3	4
р	4	4	4	3	2	3	3	4
а	5	5	5	4	3	3	4	3



Применение к исправлению опечаток

- Как исправлять опечатки с помощью расстояния Левенштейна?

Применение к исправлению опечаток

- Как исправлять опечатки с помощью расстояния Левенштейна?
- Наивный подход: пройти по словарю, подсчитать расстояние до каждого слова, выбрать ближайшее.

Применение к исправлению опечаток

- Как исправлять опечатки с помощью расстояния Левенштейна?
- Наивный подход: пройти по словарю, подсчитать расстояние до каждого слова, выбрать ближайшее.
- Нельзя: словари большие (агглютинативные языки — сотни тысяч слов), а расстояние считается медленно:

Применение к исправлению опечаток

- Как исправлять опечатки с помощью расстояния Левенштейна?
- Наивный подход: пройти по словарю, подсчитать расстояние до каждого слова, выбрать ближайшее.
- Нельзя: словари большие (агглютинативные языки — сотни тысяч слов), а расстояние считается медленно:
  - Словарь состоит из 119774 слов (стандартный словарь в /usr/share/dic в UNIX-системах),

Применение к исправлению опечаток

- Как исправлять опечатки с помощью расстояния Левенштейна?
- Наивный подход: пройти по словарю, подсчитать расстояние до каждого слова, выбрать ближайшее.
- Нельзя: словари большие (агглютинативные языки — сотни тысяч слов), а расстояние считается медленно:
  - Словарь состоит из 119774 слов (стандартный словарь в /usr/share/dic в UNIX-системах),
  - Время расчёта (Intel Atom 1.67GHz):

Python3	Python3 (Levenshtein)	C++	C++ (-O3)
144.06	0.51	4.01	1.05

Применение к исправлению опечаток

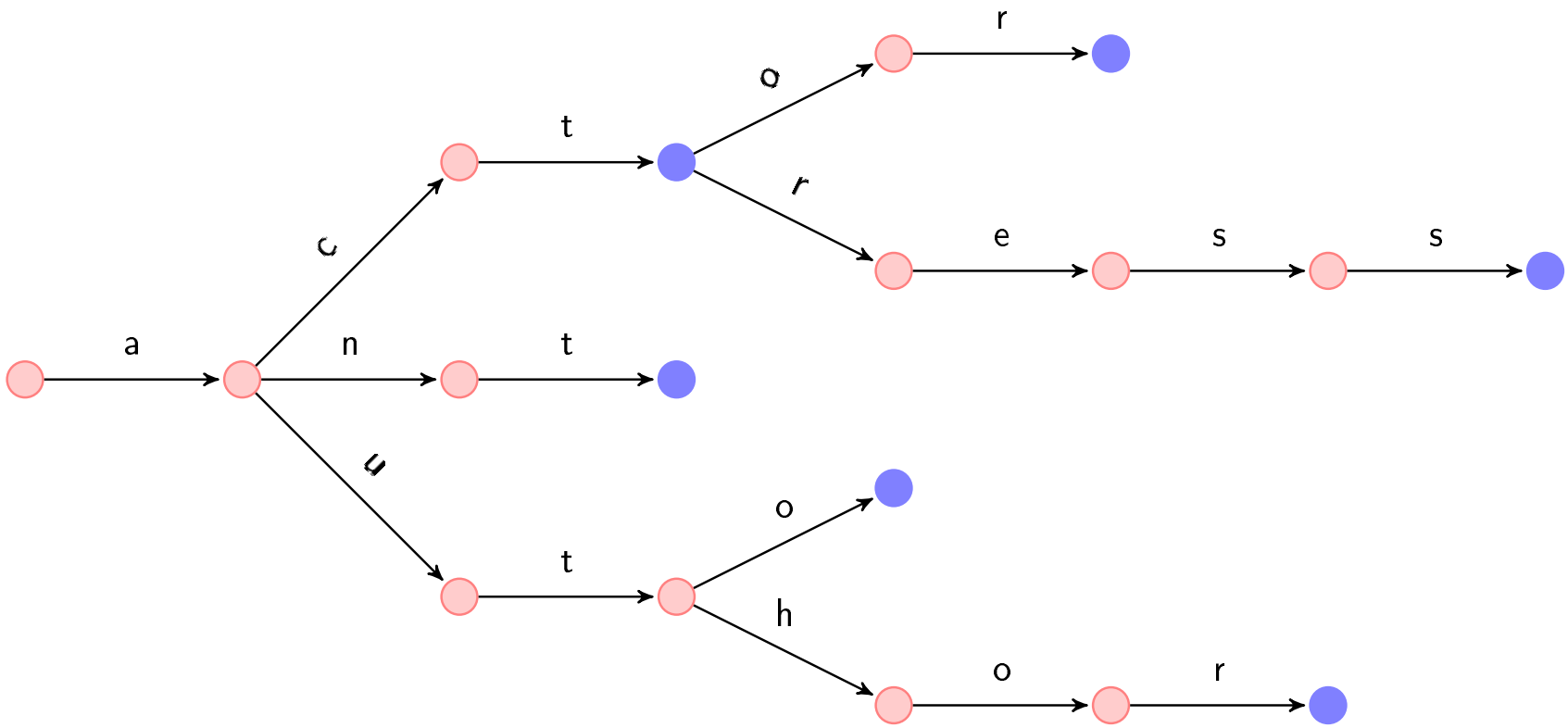
- Как исправлять опечатки с помощью расстояния Левенштейна?
- Наивный подход: пройти по словарю, подсчитать расстояние до каждого слова, выбрать ближайшее.
- Нельзя: словари большие (агглютинативные языки — сотни тысяч слов), а расстояние считается медленно:
  - Словарь состоит из 119774 слов (стандартный словарь в /usr/share/dic в UNIX-системах),
  - Время расчёта (Intel Atom 1.67GHz):

Python3	Python3 (Levenshtein)	C++	C++ (-O3)
144.06	0.51	4.01	1.05

- Менее наивный подход: хранить словарь в сжатом виде, допускающем эффективный приближённый поиск.

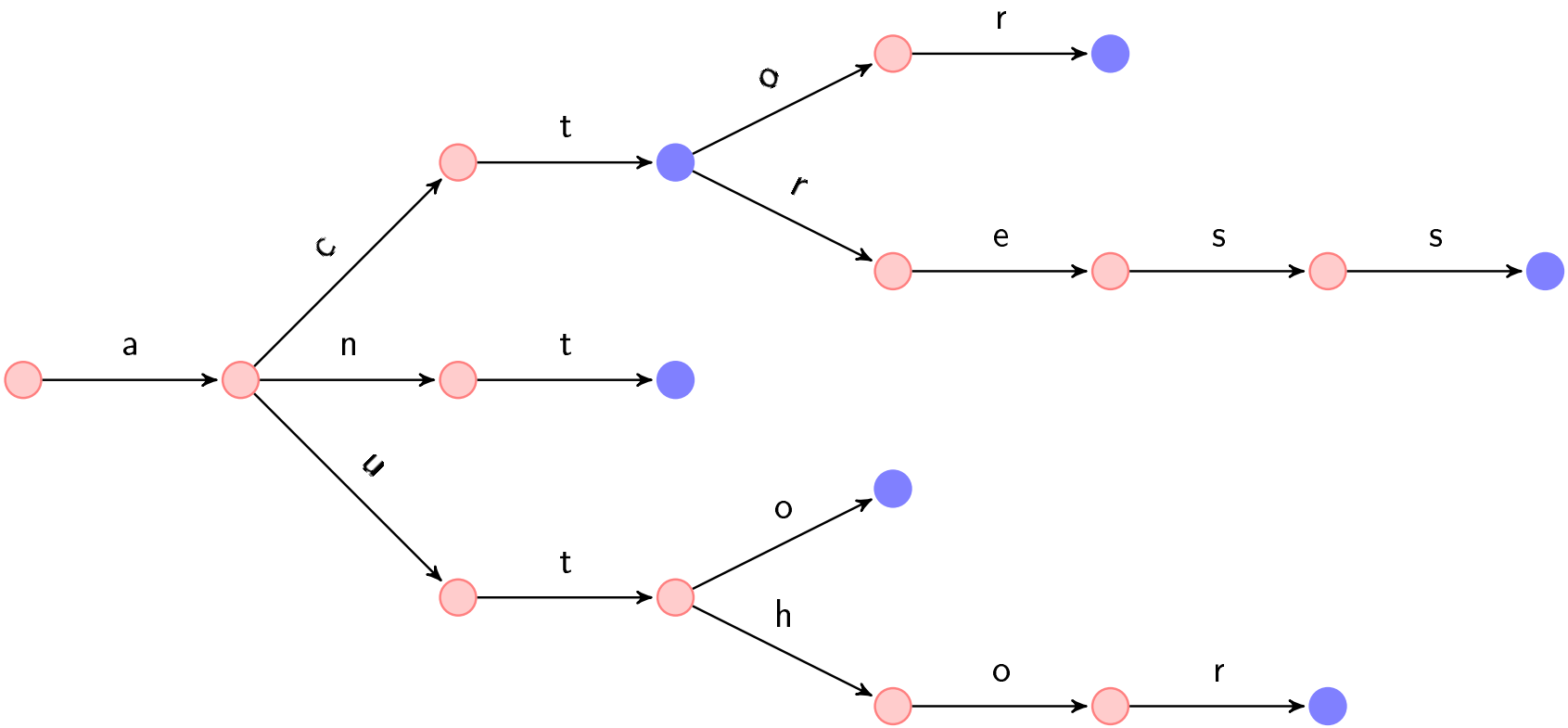
Приближённый поиск

- Словарь хранят в виде префиксного бора:



Приближённый поиск

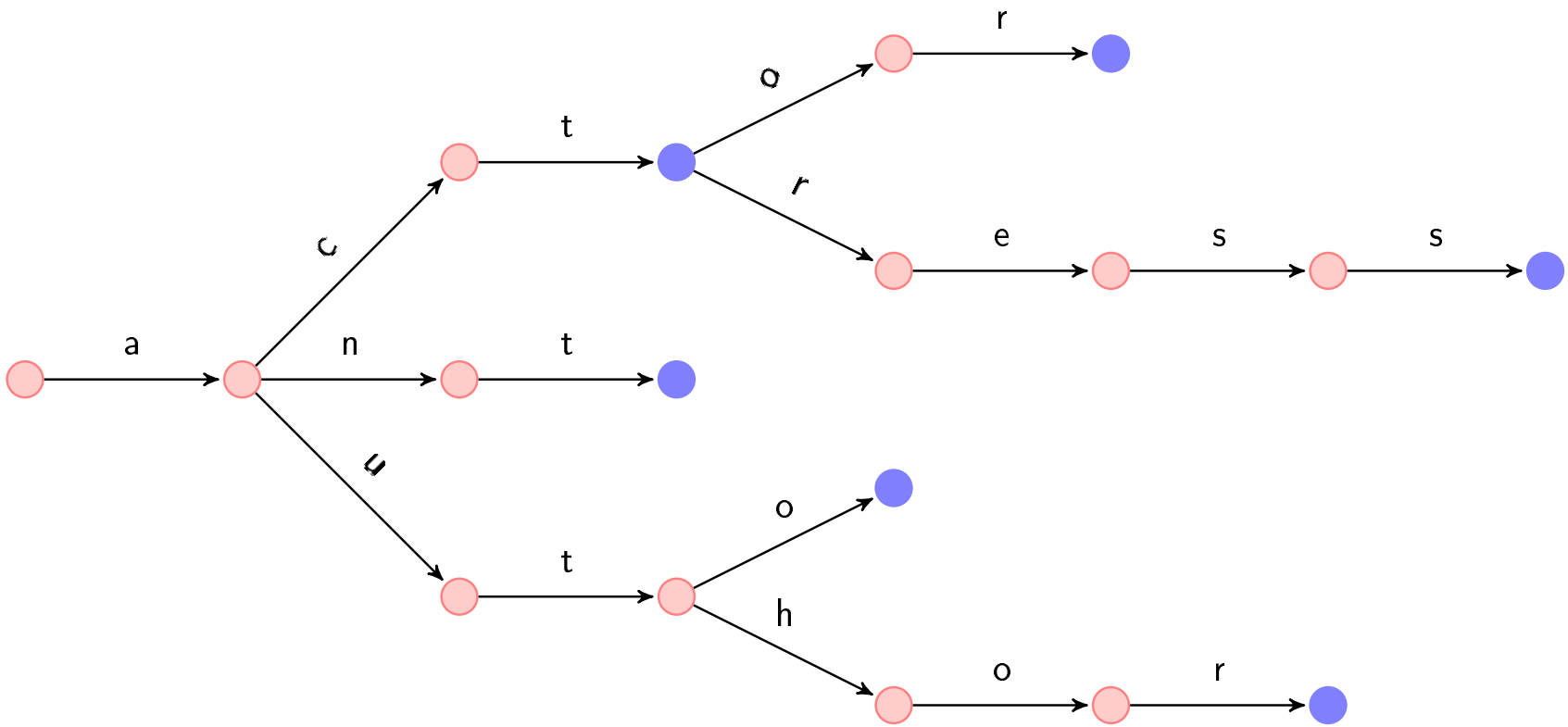
- Словарь хранят в виде префиксного бора:



- Искать слово в таком автомате легко (просто идём по рёбрам)...

Приближённый поиск

- Словарь хранят в виде префиксного бора:



- Искать слово в таком автомате легко (просто идём по рёбрам)...
- Приближённый поиск: разрешаем не более k отклонений (замен, лишних букв или удалений) от пути по слову, в процессе работы считаем отклонения.

Взвешенное расстояние Левенштейна

- Не всегда расстояние Левенштейна приводит к оптимальному выравниванию:

Взвешенное расстояние Левенштейна

- Не всегда расстояние Левенштейна приводит к оптимальному выравниванию:
- $d(loup, lobo) = 2$:

l	o	u	p
$\|$	$\|$	$\|$	$\|$
l	o	b	o

Взвешенное расстояние Левенштейна

- Не всегда расстояние Левенштейна приводит к оптимальному выравниванию:
- $d(loup, lobo) = 2$:

l	o	u	p
l	o	b	o

- Естественное выравнивание даёт $d = 3$:

l	o	u	p	$\varnothing$
l	o	$\varnothing$	b	o

Взвешенное расстояние Левенштейна

- Не всегда расстояние Левенштейна приводит к оптимальному выравниванию:
- $d(loup, lobo) = 2$:

l	o	u	p
l	o	b	o

- Естественное выравнивание даёт $d = 3$:

l	o	u	p	$\varnothing$
l	o	$\varnothing$	b	o

- Надо присвоить различным операциям веса, зависящие от заменяемых/удаляемых/вставляемых символов.

Взвешенное расстояние Левенштейна

- Не всегда расстояние Левенштейна приводит к оптимальному выравниванию:
- $d(loup, lobo) = 2$:

l	o	u	p
l	o	b	o

- Естественное выравнивание даёт $d = 3$:

l	o	u	p	$\varnothing$
l	o	$\varnothing$	b	o

- Надо присвоить различным операциям веса, зависящие от заменяемых/удаляемых/вставляемых символов.
- Алгоритм вычисления расстояния от этого не изменится (при естественных ограничениях на веса).

Поиск слов-кандидатов

- Поиск по расстоянию Левенштейна может вернуть несколько кандидатов:

  *стадо *свией бросилось в море и утонуло*

Поиск слов-кандидатов

- Поиск по расстоянию Левенштейна может вернуть несколько кандидатов:

  *стадо *свией бросилось в море и утонуло*
- **свией* $\mapsto$ *сваей*, *свиней*, *своей*, *свей*, ...

Поиск слов-кандидатов

- Поиск по расстоянию Левенштейна может вернуть несколько кандидатов:

  *стадо *свией бросилось в море и утонуло*
- **свией* $\mapsto$ *сваей*, *свиней*, *своей*, *свей*, ...
- Нужно определить, какое слово наиболее вероятно в данном контексте.

Поиск слов-кандидатов

- Поиск по расстоянию Левенштейна может вернуть несколько кандидатов:

  *стадо *свией бросилось в море и утонуло*
- **свией* $\mapsto$ *сваей*, *свиней*, *своей*, *свей*, ...
- Нужно определить, какое слово наиболее вероятно в данном контексте.
- То есть какое из предложений-исправлений имеет наибольшую вероятность?

Поиск слов-кандидатов

- Поиск по расстоянию Левенштейна может вернуть несколько кандидатов:

  *стадо *свией бросилось в море и утонуло*
- **свией* $\mapsto$ *сваей*, *свиней*, *своей*, *свей*, ...
- Нужно определить, какое слово наиболее вероятно в данном контексте.
- То есть какое из предложений-исправлений имеет наибольшую вероятность?
- Математически:

$$
\begin{array}{rcl}
\widehat{\mathbf{t}} & = & \operatorname{argmax}_{\mathbf{t}} p(\mathbf{t}|\mathbf{w}) \\
\widehat{\mathbf{t}} & — & \text{наилучшее исправление,} \\
\mathbf{t} & = & t_1 \dots t_n, \\
\mathbf{t} & — & \text{одно из возможных исправлений,} \\
\mathbf{w} & = & w_1 \dots w_n, \\
\mathbf{w} & — & \text{исходное предложение,}
\end{array}
$$

Модель исправления

- Вероятностная постановка задачи:

$$\widehat{\mathbf{t}} = \operatorname{argmax}_{\mathbf{t}} p(\mathbf{t}|\mathbf{w})$$

Модель исправления

- Вероятностная постановка задачи:

$$\widehat{\mathbf{t}} = \operatorname{argmax}_{\mathbf{t}} p(\mathbf{t}|\mathbf{w})$$

- По формуле Байеса:

$$\widehat{\mathbf{t}} = \operatorname{argmax}_{\mathbf{t}} p(\mathbf{t}|\mathbf{w}) = \operatorname{argmax}_{\mathbf{t}} \frac{p(\mathbf{w}|\mathbf{t})p(\mathbf{t})}{p(\mathbf{w})}$$

Модель исправления

- Вероятностная постановка задачи:

$$\widehat{\mathbf{t}} = \operatorname{argmax}_{\mathbf{t}} p(\mathbf{t}|\mathbf{w})$$

- По формуле Байеса:

$$\widehat{\mathbf{t}} = \operatorname{argmax}_{\mathbf{t}} p(\mathbf{t}|\mathbf{w}) = \operatorname{argmax}_{\mathbf{t}} \frac{p(\mathbf{w}|\mathbf{t})p(\mathbf{t})}{p(\mathbf{w})}$$

- $p(\mathbf{w})$ всё время одинаковая:

$$\widehat{\mathbf{t}} = \operatorname{argmax}_{\mathbf{t}} p(\mathbf{w}|\mathbf{t})p(\mathbf{t})$$

Модель исправления

- Вероятностная постановка задачи:

$$\widehat{\mathbf{t}} = \operatorname{argmax}_{\mathbf{t}} p(\mathbf{t}|\mathbf{w})$$

- По формуле Байеса:

$$\widehat{\mathbf{t}} = \operatorname{argmax}_{\mathbf{t}} p(\mathbf{t}|\mathbf{w}) = \operatorname{argmax}_{\mathbf{t}} \frac{p(\mathbf{w}|\mathbf{t})p(\mathbf{t})}{p(\mathbf{w})}$$

- $p(\mathbf{w})$ всё время одинаковая:

$$\widehat{\mathbf{t}} = \operatorname{argmax}_{\mathbf{t}} p(\mathbf{w}|\mathbf{t})p(\mathbf{t})$$

- $p(\mathbf{t}) = p(t_1 \dots t_n)$ — вероятность увидеть текст $t_1 \dots t_n$.
- $p(\mathbf{w}|\mathbf{t}) = p(w_1 \dots w_n|t_1 \dots t_n)$ — вероятность, что в результате опечаток текст $t_1 \dots t_n$ превратится в $w_1 \dots w_n$.

Вероятностная модель текста

- $p(\mathbf{t}) = p(t_1 \dots t_n)$ — вероятность увидеть текст $t_1 \dots t_n$.
- Мы пока не будем вводить точную формулу.

Вероятностная модель текста

- $p(\mathbf{t}) = p(t_1 \dots t_n)$ — вероятность увидеть текст $t_1 \dots t_n$.
- Мы пока не будем вводить точную формулу.
- Она учитывает статистику совместной встречаемости:

Фраза	Число вхождений
стадо свиней	15
свиней бросилось	4
стадо своей	3
своей бросилось	1

Вероятностная модель текста

- $p(\mathbf{t}) = p(t_1 \dots t_n)$ — вероятность увидеть текст $t_1 \dots t_n$.
- Мы пока не будем вводить точную формулу.
- Она учитывает статистику совместной встречаемости:

Фраза	Число вхождений
стадо свиней	15
свиней бросилось	4
стадо своей	3
своей бросилось	1

- Тогда

$$p(\dots \textit{стадо свиней бросилось} \dots) \gg p(\dots \textit{стадо своей бросилось} \dots)$$

- Это *энграммная модель языка*.

Вероятностная модель текста

- $p(\mathbf{t}) = p(t_1 \dots t_n)$ — вероятность увидеть текст $t_1 \dots t_n$.
- Мы пока не будем вводить точную формулу.
- Она учитывает статистику совместной встречаемости:

Фраза	Число вхождений
стадо свиней	15
свиней бросилось	4
стадо своей	3
своей бросилось	1

- Тогда

$$p(\dots \textit{стадо свиней бросилось} \dots) \gg p(\dots \textit{стадо своей бросилось} \dots)$$

- Это *энграммная модель языка*.
- Ещё можно посмотреть на последовательность морфологических категорий и по ним оценить корректность (тоже с помощью энграммной модели).

Модель исправления

- $p(\mathbf{w}|\mathbf{t}) = p(w_1 \dots w_n | t_1 \dots t_n)$ — вероятность, что в результате опечаток текст $t_1 \dots t_n$ превратится в $w_1 \dots w_n$.

Модель исправления

- $p(\mathbf{w}|\mathbf{t}) = p(w_1 \dots w_n|t_1 \dots t_n)$ — вероятность, что в результате опечаток текст $t_1 \dots t_n$ превратится в $w_1 \dots w_n$.
- В каждом слове опечатки происходят независимо:

$$p(w_1 \dots w_n|t_1 \dots t_n) = p(w_1|t_1) \dots p(w_n|t_n)$$

- $p(w_i|t_i)$ — вероятность получить слово w_i из t_i в результате опечатки.

Модель исправления

- $p(\mathbf{w}|\mathbf{t}) = p(w_1 \dots w_n|t_1 \dots t_n)$ — вероятность, что в результате опечаток текст $t_1 \dots t_n$ превратится в $w_1 \dots w_n$.
- В каждом слове опечатки происходят независимо:

$$p(w_1 \dots w_n|t_1 \dots t_n) = p(w_1|t_1) \dots p(w_n|t_n)$$

- $p(w_i|t_i)$ — вероятность получить слово w_i из t_i в результате опечатки.
- Чем расстояние Левенштейна больше, тем вероятность меньше.
- Например,

$$p(w_i|t_i) = Ce^{-\alpha d(w_i,t_i)}, \text{ если } d(w_i, t_i) \leqslant 2, \text{ иначе } 0$$

- C — нормирующая константа (чтобы вероятности суммировались к 1). α — коэффициент вероятности ошибки.

Уточнение модели опечаток

- Опечатки не сводятся к расстоянию Левенштейна:
  - Вставка и удаление пробелов/дефисов (кому-то $\mapsto$ *комуто*, *кому то*).

Уточнение модели опечаток

- Опечатки не сводятся к расстоянию Левенштейна:
  - Вставка и удаление пробелов/дефисов (кому-то $\mapsto$ *комуто*, *кому то*).
  - Перестановка соседних символов.

Уточнение модели опечаток

- Опечатки не сводятся к расстоянию Левенштейна:
  - Вставка и удаление пробелов/дефисов (кому-то $\mapsto$ *комуто*, *кому то*).
  - Перестановка соседних символов.
  - Удвоение/отсутствие удвоения символа.

Уточнение модели опечаток

- Опечатки не сводятся к расстоянию Левенштейна:
  - Вставка и удаление пробелов/дефисов (кому-то $\mapsto$ *комуто*, *кому то*).
  - Перестановка соседних символов.
  - Удвоение/отсутствие удвоения символа.
  - Просторечие:

    $$\textit{вообще} \mapsto *\textit{ваще},\ \textit{только} \mapsto *\textit{тока}.$$

  - Неформальные сокращения (Twitter):

    $$\textit{you} \mapsto U,\ \textit{to} \mapsto 2,\ \textit{for} \mapsto 4.$$

Уточнение модели опечаток

- Опечатки не сводятся к расстоянию Левенштейна:
  - Вставка и удаление пробелов/дефисов (кому-то $\mapsto$ *комуто*, *кому то*).
  - Перестановка соседних символов.
  - Удвоение/отсутствие удвоения символа.
  - Просторечие:

    $$\textit{вообще} \mapsto *\textit{ваще}, \ \textit{только} \mapsto *\textit{тока}.$$

  - Неформальные сокращения (Twitter):

    $$\textit{you} \mapsto U, \ \textit{to} \mapsto 2, \ \textit{for} \mapsto 4.$$

- Наличие опечатки зависит от позиции в слове (суффикс прилагательного/глагола).
- орфографические ошибки происходят на фонетическом уровне:

  $$\textit{караван} \mapsto *\textit{кораван}$$

Уточнение модели опечаток

- Опечатки не сводятся к расстоянию Левенштейна:
  - Вставка и удаление пробелов/дефисов (кому-то $\mapsto$ *комуто*, *кому то*).
  - Перестановка соседних символов.
  - Удвоение/отсутствие удвоения символа.
  - Просторечие:

    $$\textit{вообще} \mapsto *\textit{ваще}, \ \textit{только} \mapsto *\textit{тока}.$$

  - Неформальные сокращения (Twitter):

    $$\textit{you} \mapsto U, \ \textit{to} \mapsto 2, \ \textit{for} \mapsto 4.$$

- Наличие опечатки зависит от позиции в слове (суффикс прилагательного/глагола).
- орфографические ошибки происходят на фонетическом уровне:

  $$\textit{караван} \mapsto *\textit{кораван}$$

- В зависимости от источника — ошибки разной природы (собственно опечатки, просторечие, орфографические ошибки, ошибки распознавания).

Словарные ошибки

- Ошибки могут приводить к словарным словам:

  На голову его величества была возложена . . .

  корова? ворона? борона?

Словарные ошибки

- Ошибки могут приводить к словарным словам:

  На голову его величества была возложена . . .

  корова? ворона? борона?

- Следствие: если слово из словаря, это не значит, что в нём нет опечатки.

Словарные ошибки

- Ошибки могут приводить к словарным словам:

  На голову его величества была возложена . . .

  корова? ворона? борона?

- Следствие: если слово из словаря, это не значит, что в нём нет опечатки.
- А если не из словаря, то не значит, что есть (неологизмы, имена собственные, редкие слова).

Словарные ошибки

- Ошибки могут приводить к словарным словам:

  На голову его величества была возложена . . .

  корова? ворона? борона?

- Следствие: если слово из словаря, это не значит, что в нём нет опечатки.
- А если не из словаря, то не значит, что есть (неологизмы, имена собственные, редкие слова).
- Частый источник опечаток: омофоны/паронимы:

  умолять / умалять

  компания / кампания

  разрядить / разредить

- Как следствие, надо предполагать, что опечатка может иметься в каждом слове.

Словарные ошибки

- Ошибки могут приводить к словарным словам:

  На голову его величества была возложена . . .

  корова? ворона? борона?

- Следствие: если слово из словаря, это не значит, что в нём нет опечатки.
- А если не из словаря, то не значит, что есть (неологизмы, имена собственные, редкие слова).
- Частый источник опечаток: омофоны/паронимы:

  умолять / умалять

  компания / кампания

  разрядить / разредить

- Как следствие, надо предполагать, что опечатка может иметься в каждом слове.
- Вероятность опечатки зависит от слова (словарное/несловарное, собственное/нарицательное, часть речи, длина слова).

Подбор предложений-кандидатов

Стадо	свией	бросилось	со	скалы	в	море
стадо	свиней своей сваей	бросилось	со сто до то	скалы шкалы скулы	в во с	море мире горе торе

Подбор предложений-кандидатов

Стадо	свией	бросилось	со	скалы	в	море
стадо	свиней своей сваей	бросилось	со сто до то	скалы шкалы скулы	в во с	море мире горе торе

- Возможные предложения:

  стадо свиней бросилось со скалы в море

  стадо сваей бросилось со скалы в море

  стадо своей бросилось со скалы в море

  ...

  стадо своей бросилось со скулы в хоре

Подбор предложений-кандидатов

Стадо	свией	бросилось	со	скалы	в	море
стадо	свиней своей сваей	бросилось	со сто до то	скалы шкалы скулы	в во с	море мире горе торе

- Возможные предложения:
  стадо свиней бросилось со скалы в море
  стадо сваей бросилось со скалы в море
  стадо своей бросилось со скалы в море
  ...
  стадо своей бросилось со скулы в хоре
- Вариантов “исправлений" слишком много, нужно поддерживать только небольшое количество лучших.

Онлайн-материалы

- `http://norvig.com/spell-correct.html` — страница Питера Норвига. Простейшая работающая программа для исправления опечаток из 21 строчки (Python 2.5).
- Ещё много материалов: `http://norvig.com`
- `http://dialog-21.ru/evaluation/2016/spelling_correction` — соревнование по исправлению опечаток для русского языка.