МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(национальный исследовательский университет)

Распределенное программно-информационное обеспечение статистической модели перевода естественных языков

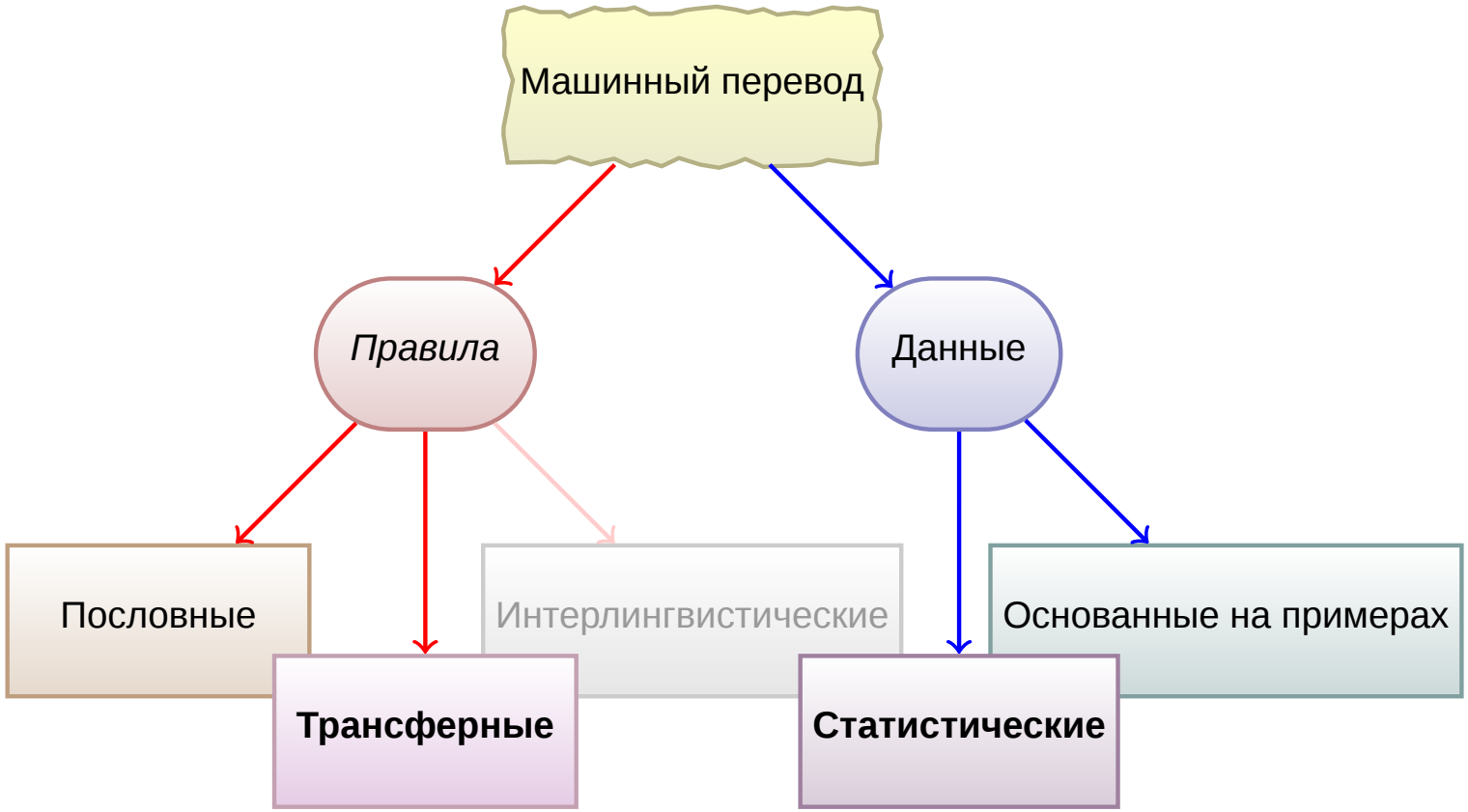
Выполнил студент группы 08-606
Никитин Илья Константинович
Научный руководитель
ассистент кафедры 806
Гаврилов Евгений Сергеевич

Содержание

Для чего нужен машинный перевод?

- бытовой перевод:
  - книги,
  - переписка;
- поиск в Интернете на разных языках (внутри поисковых алгоритмов и дополнительная функция для пользователя);
- перевод научных публикаций с других языков;
- применения достижений в других областях:
  - автоматическое реферирование,
  - распознавание речи,
  - распознавание последовательностей аминокислот (ДНК).

Основные методы машинного перевода



Модель зашумленного канала (1)

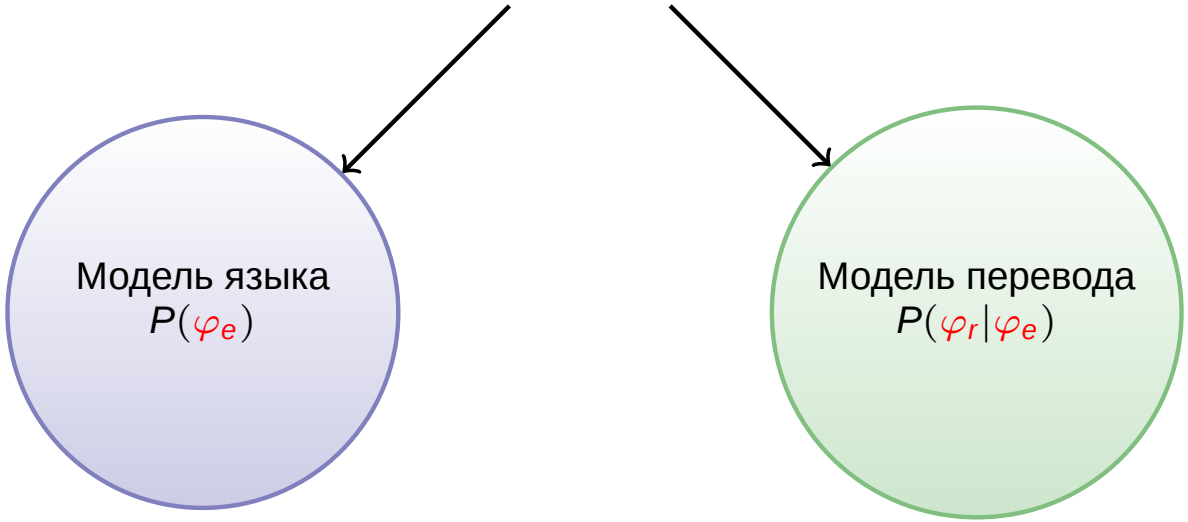
1. Пусть φ_r — фраза оригинала (русская).
2. Требуется найти φ_e — фразу перевода (английскую).

Максимизировать $P(\varphi_e|\varphi_r)$.

$$P(\varphi_e|\varphi_r) = \frac{(P(\varphi_e) \cdot P(\varphi_r|\varphi_e))}{P(\varphi_r)} \Rightarrow$$

$$\varphi_{eg} = \arg\max_{\cup\varphi_e} P(\varphi_e|\varphi_r) = \arg\max_{\cup\varphi_e} (P(\varphi_e) \cdot P(\varphi_r|\varphi_e))$$

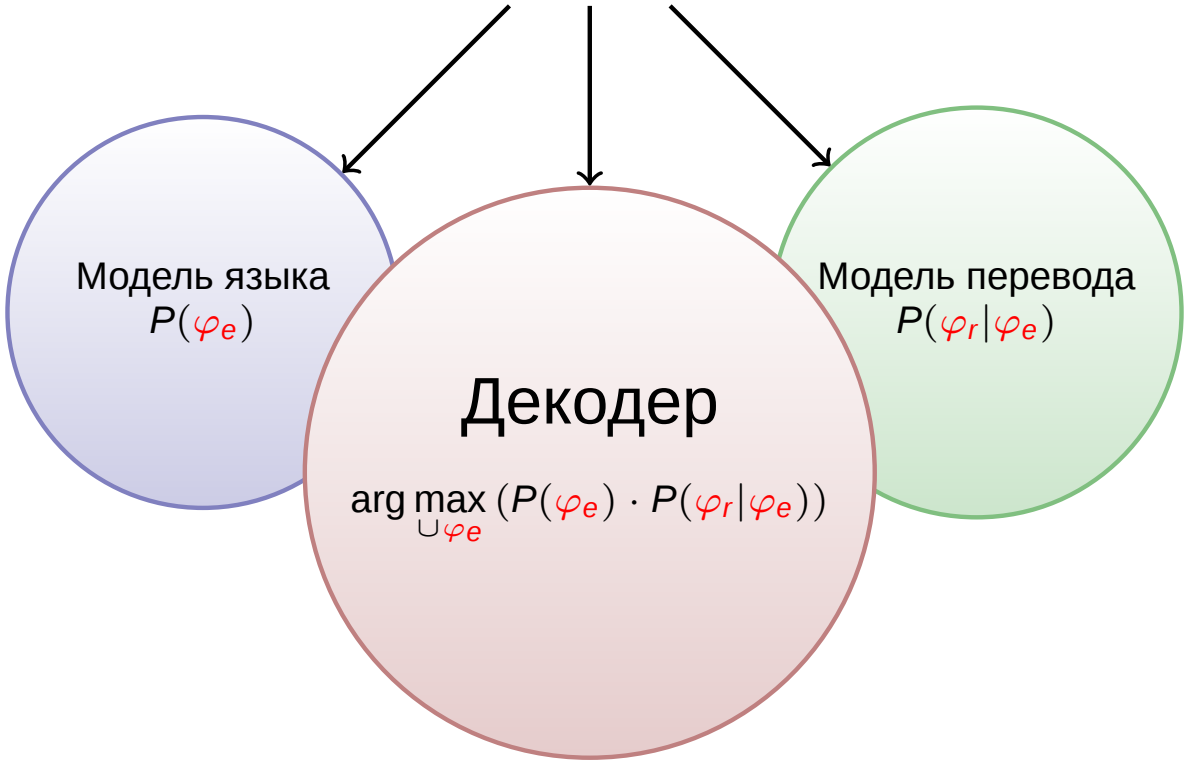
Статистическая система машинного перевода



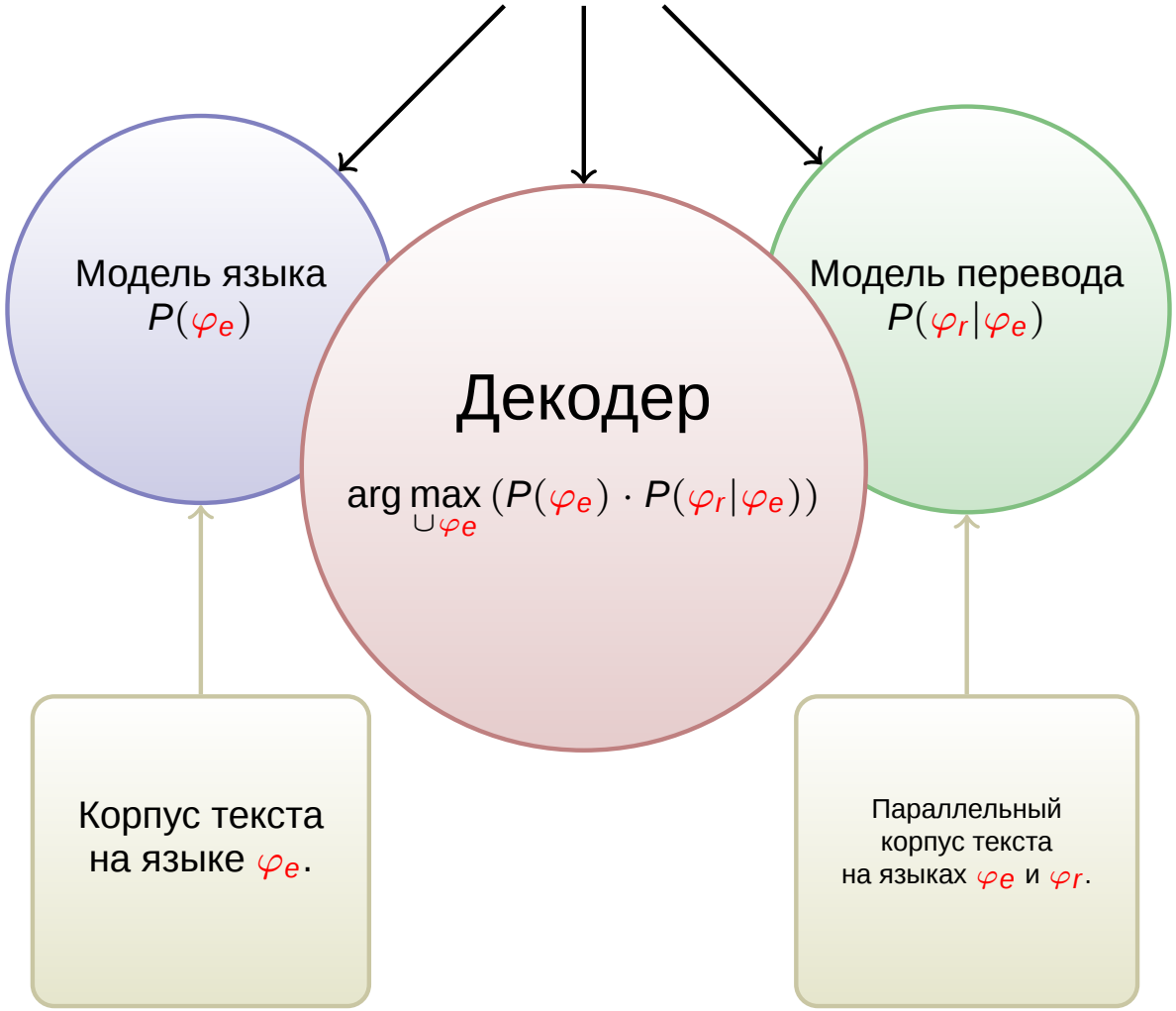
$$\arg\max_{\cup\varphi_e} P(\varphi_e|\varphi_r) = \arg\max_{\cup\varphi_e} (P(\varphi_e) \cdot P(\varphi_r|\varphi_e))$$

- φ_e — фраза перевода (английская);
- φ_r — фраза оригинала (русская).

Статистическая система машинного перевода



Статистическая система машинного перевода



Модель языка

- Правильный порядок слов.
- Вычисляется с помощью n-грамм слов. Пример для 3-грамм:

$$\varphi = (\omega_1, \omega_2, \omega_3, \omega_4, \ldots, \omega_l) \Rightarrow \begin{cases} (\omega_1, & \omega_2, & \omega_3); \\ (\omega_2, & \omega_3, & \omega_4); \\ \vdots & \vdots & \vdots \\ (\omega_{l-2}, & \omega_{l-1}, & \omega_l). \end{cases}$$

- Вычисляется по формуле:

$$P(\varphi) = P(\omega_1 \ldots \omega_l) = \prod_{i=0}^{i=l+n-1} P'(\omega_i | \omega_{i-1} \ldots \omega_{i-n+1}).$$

Модель перевода (1)

- Вводим выравнивание для пары предложений Π_e, Π_r.
- Для выравнивания нужны вероятности лексического перевода $\omega_e \rightarrow \omega_r$.
- Для вероятности лексического перевода нужны выравнивания.
- Проблема «курицы и яйца».

Модель перевода (2)

Для оценки вероятности лексического перевода $\longrightarrow$
EM-алгоритм (Витерби):

- инициализируем параметры модели (одинаковыми значениями, на первой итерации);
- оценим вероятности отсутствующей информации;
- оценим параметры модели на основании новой информации;
- перейдем к следующей итерации.

∃ Отличия от других систем

Система используется для перевода научно-технической литературы.

Слова → *n*-грамы

⇐ Устойчивые формальные выражения в научных текстах.

Выравнивание по круппным группам *n*-грам

⇐ прямой порядок слов;

⇐ стереотипная структура предложений.

Модели низких порядков

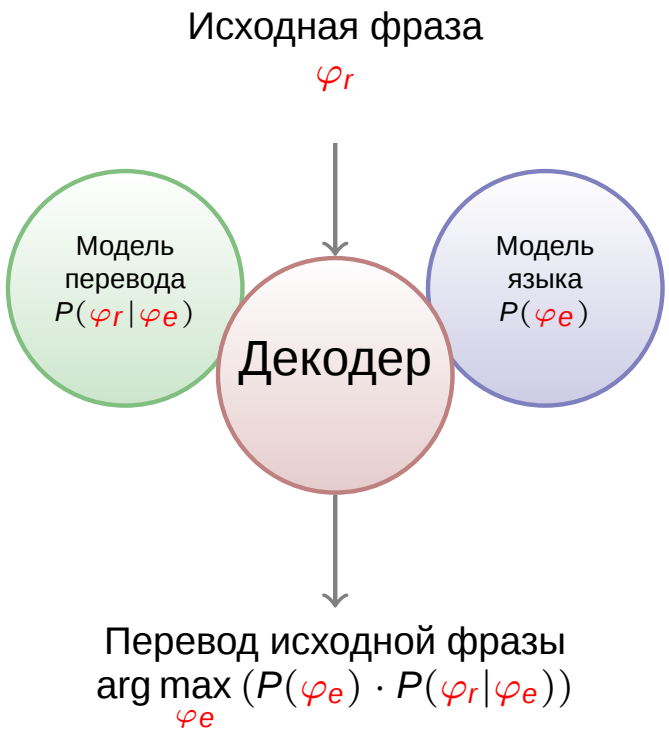
⇐ важность локального порядка слов;

⇐ фертильности и вероятностной грамматики могут его разрушить.

Декодер

Среди всех возможных вариантов перевода выбрать правильный:

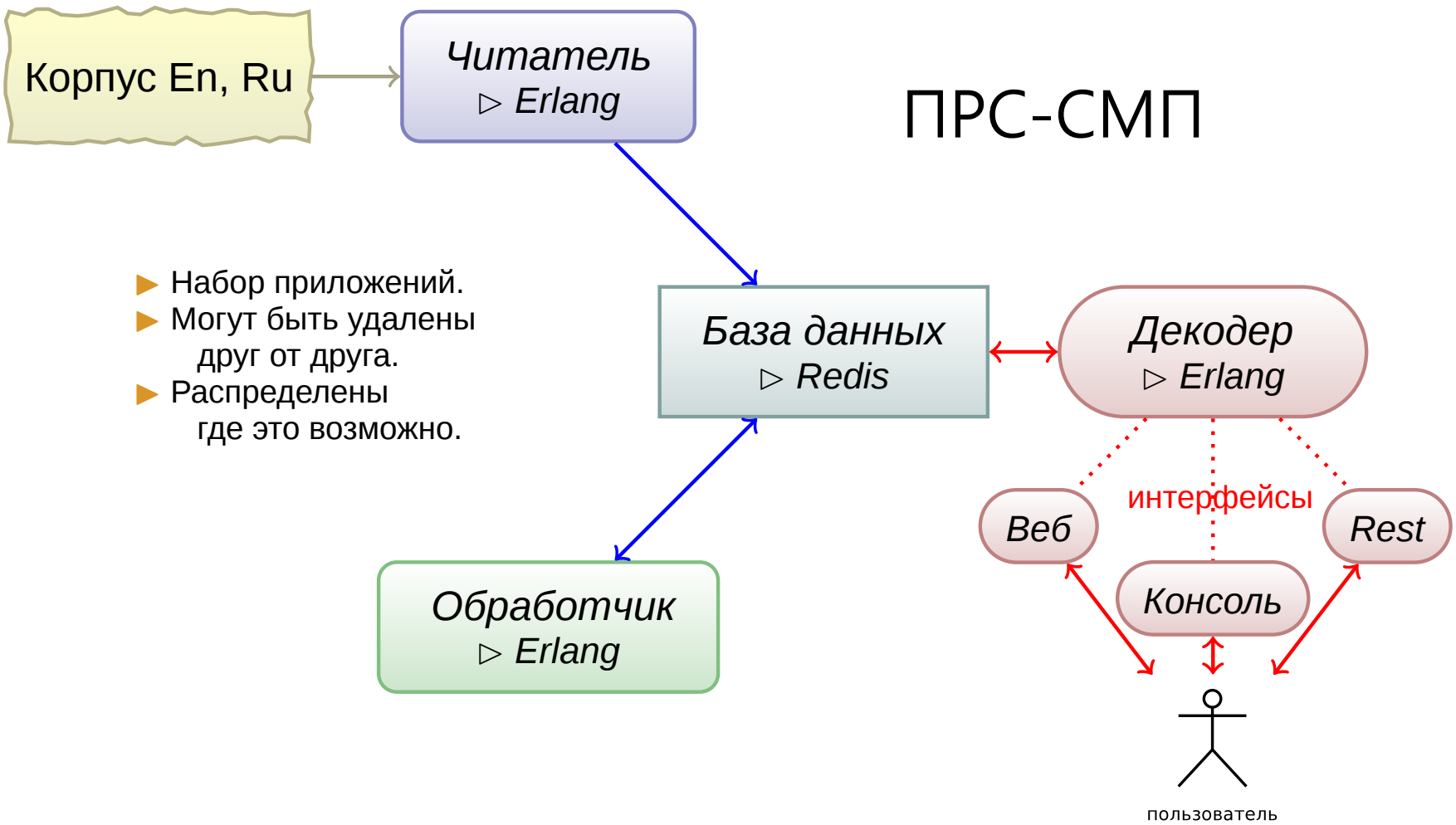
- полный перебор;
- A*:
  - стековый поиск,
  - многостековый поиск;
- **жадный инкрементный поиск**;
- сведение к обобщенной задаче коммивояжера.



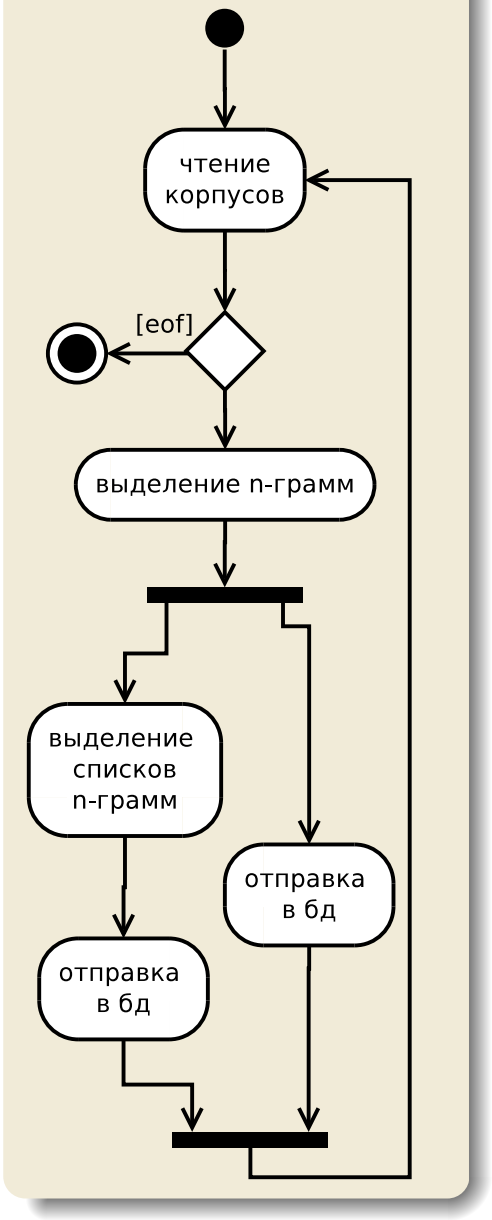
Жадный инкрементный поиск

- простой и быстрый поиск;
- «плохой» вариант перевода получаем сразу;
- последовательно применяя набор операций можем улучшить перевод;
  - изменить перевод слова *(группы слов, n-граммы)*,
  - удалить слово *(группу слов, n-грамму)*,
  - поменять слова местами *(группы слов, n-граммы)*;
- можно делать отсечку по времени;
- можем сразу оценить модель языка большой фразы.

∃ Из чего состоит система

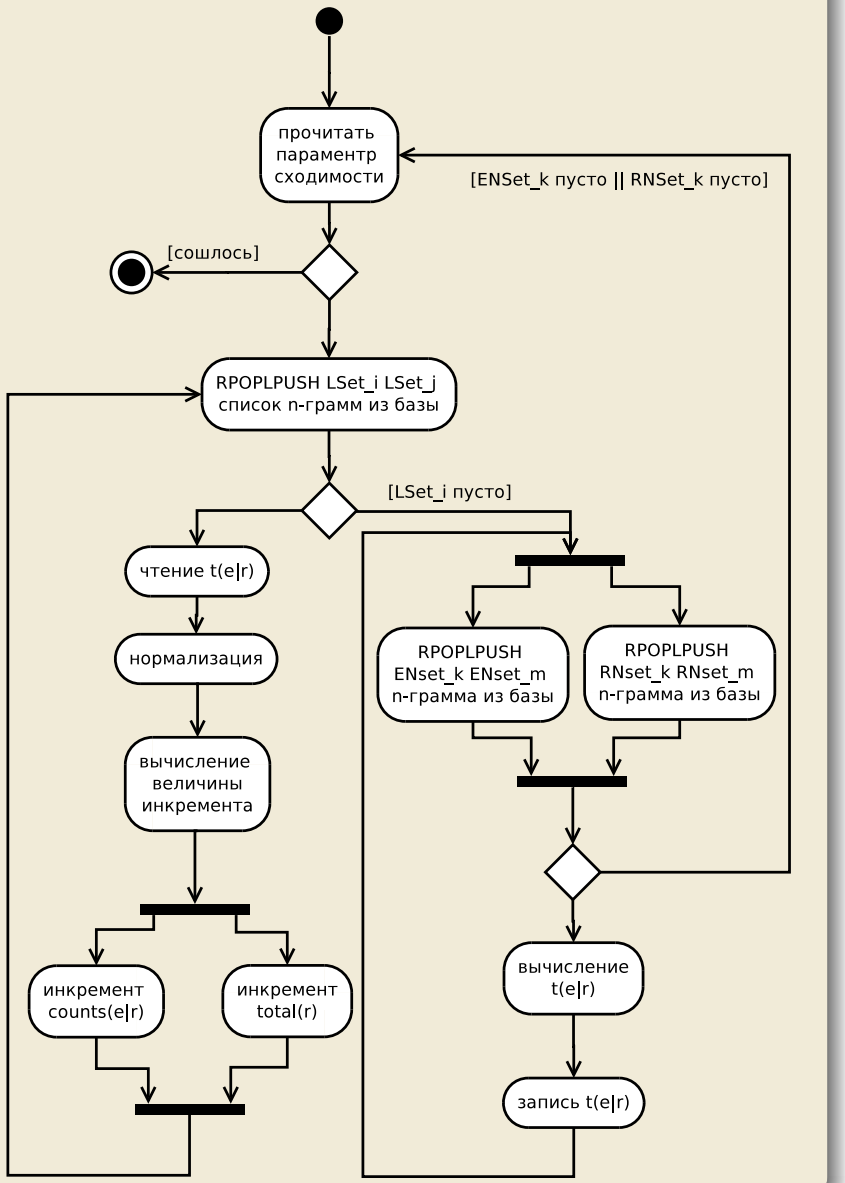


Читатель

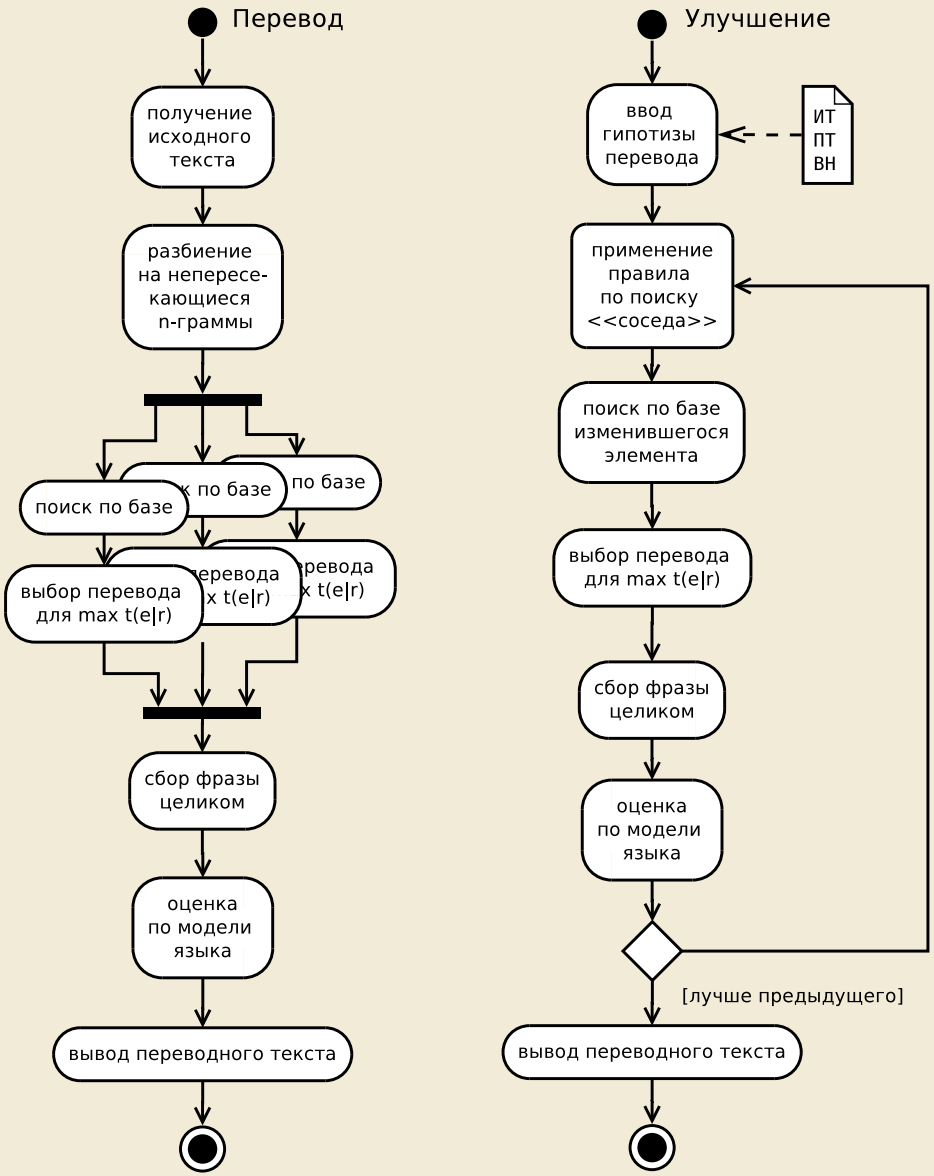


$$N_{\text{чит.}} < N_{\text{обр.}}$$

Обработчик



Декодер



- жадный инкрементный поиск;
- два режима работы:
  - перевода,
  - улучшения.
- пошаговый веб-интерфейс;
- потоковый RESTful-сервис;
- пошаговый консольный интерфейс.

∃ Примеры (1)

Оригинал

... adopted at the 81st plenary meeting ...

Переводчик

... принята на 81-м пленарном заседании ...

Система

... принята без голосования на 81 пленарном заседании в Брюсселе ...

∃ Примеры (2)

Оригинал

It will be instructive to exhibit Euclid’s algorithm here.

Переводчик

Думаю, имеет смысл привести здесь описание этого алгоритма.

Система

Будет поучительно выставить алгоритм Евклида здесь.

∃ Примеры (3)

Оригинал

Many years have passed since the author wrote most of the comments above ...

Переводчик

Со времени первого написания автором большинства приведенных выше комментариев утекло много воды ...

Система

Много лет прошло с тех пор, автор написал большую часть комментариев выше ...

Оценка перевода с использованием метрики BLEU

- BLEU — Bilingual Evaluation Understudy
- Численная оценка качества перевода.
- Нужен перевод, выполненный человеком.
- Показывает величину близости к «человеческому» переводу.
- Чем меньше величина, тем лучше.
- Сравнивались:
  - ПРС-СМП;
  - системы построенная на основе Moses.

Система	BLEU
ПРС-СМП (1)	0.243
ПРС-СМП (100)	0.209
Moses (IBM 3)	0.201
Moses (IBM 5)	0.173

Оценка скорости обучения

Процессор: Intel Core2 Duo, 1 ядро 64 бит, ОП 4Гб, ФС:ext4

Система	Время, ч
ПРС-СМП (1)	≈ 5
Moses (GIZA++)	≈ 25
Chaski (MGIZA++)	≈ 26

Процессор: Intel Xeon E5506, 8 ядер 64 бит, ОП 10Гб, ФС:xfs

Система	Время, ч
ПРС-СМП (1)	≈ 1
Moses (GIZA++)	≈ 22
Chaski (MGIZA++)	≈ 3

Оценка скорости декодирования

Процессор: Intel Core2 Duo, 1 ядро 64 бит, ОП 4Гб, ФС:ext4

Система	Время, мкс
ПРС-СМП (1)	1132
ПРС-СМП (100)	7108124
Moses (IBM 3)	$\approx$ 10000000
Moses (IBM 5)	$\approx$ 30000000

Процессор: Intel Xeon E5506, 8 ядер 64 бит, ОП 10Гб, ФС:xfs

Система	Время, мкс
ПРС-СМП (1)	1012
ПРС-СМП (100)	1119024
Moses (IBM 3)	$\approx$ 5000000
Moses (IBM 5)	$\approx$ 6000000

Результаты

- Разработан подход:
  - быстрого обучения модели перевода для научных текстов.
- Реализована система машинного перевода:
  - многопроцессорная, распределенная;
  - только научно-техническая литература;
  - быстрое обучение;
  - быстрое (пошаговое) декодирование.

Дальнейшее развитие

Математика:

- полноценный фразовый перевод;
- синтаксический перевод;
- смешанная система перевода:
  - пара русский-английский,
  - морфологический анализ.
- опробовать более точные методы поиска.

Архитектура и реализация:

- использовать пословное сжатие при хранении в БД;
- переписать обработчика на Си с libevent;
- libevent для RESTful-сервиса декодера:
  - 1 млн. одновременных соединений
- попробовать Redis $\rightarrow$ leveldb.

Результаты

- Разработан подход:
  - быстрого обучения модели перевода для научных текстов.
- Реализована система машинного перевода:
  - многопроцессорная, распределенная;
  - только научно-техническая литература;
  - быстрое обучение;
  - быстрое (пошаговое) декодирование.

Приложения-подробности

интересные слайды,
которые не вошли в саму презентацию

Модель языка

Вычисляется с помощью n-грамм слов.

$$P(\omega_1 \ldots \omega_l) = \prod_{i=0}^{i=l+n-1} P'(\omega_i | \omega_{i-1} \ldots \omega_{i-n+1})$$

- $P'(\omega_m | \omega_1 \ldots \omega_{m-1}) = K_n \cdot P(\omega_m | \omega_1 \ldots \omega_{m-1}) + \ldots + K_1 \cdot P(\omega_1) + K_0$;
- $P(\omega_1) = \frac{\textit{частота}\ (\omega_1)}{|\Theta|}$;
- $P(\omega_m | \omega_1 \ldots \omega_{m-1}) = \frac{\textit{частота}\ (\omega_1 \ldots \omega_{m-1}\omega_m)}{\textit{частота}\ (\omega_1 \ldots \omega_{m-1})}$;
- K_i — коэффициенты сглаживания $K_i > K_{i+1}$ и $\sum_{i=0}^{i=n} K_i = 1.0$.

Модель языка (адаптивные модели)

$$P(\omega_1 \dots \omega_l) = \prod_{i=0}^{i=l+n-1} P'(\omega_i | \omega_{i-1} \dots \omega_{i-n+1})$$

P' можно вычислить иначе, используя адаптивный метод сглаживания

$$P'(\omega_m | \omega_1 \dots \omega_{m-1}) = \frac{\delta + \textit{частота}\ (\omega_1 \dots \omega_m)}{\sum_i \left(\delta + \textit{частота}\ (\omega_{1_j} \dots \omega_{m_j})\right)} =$$

$$= \frac{\delta + \textit{частота}\ (\omega_1 \dots \omega_m)}{\delta \cdot V + \sum_i \left(\textit{частота}\ (\omega_{1_j} \dots \omega_{m_j})\right)}$$

- V — количество всех n-грамм в используемом корпусе;
- $\delta = 1$ — метод сглаживания Лапласа;
- $\delta \neq 1 \Rightarrow$ методы Гуда-Тьюринга, Катца, Кнезера-Нейя.

Введем обозначения

- Θ_e — «английский» текст (множество предложений);
- Θ_r — «русский» текст;
- Π_e — «английское» предложение (последовательность слов);
- Π_r — «русское» предложение;
- ω_e — «английское» слово;
- ω_r — «русское» слово;

Модель перевода (1)

Пусть $P(\Pi_e|\Pi_r)$ — вероятность некоторой строки (предложения) из e, при гипотезе перевода из r.

$$P(\Pi_e|\Pi_r) = \sum_a P(\Pi_e, a|\Pi_r);$$

a — выравнивание между отдельными словами в паре предложений. Вероятность перевода:

$$P(\Pi_e, a|\Pi_r) = \frac{\varepsilon}{(l_r+1)^{l_e}} \prod_{j=1}^{l_e} t(\omega_{ej}|\omega_{ra(j)})$$

t — это вероятность слова оригинала в позиции j при соответствующем ему слове перевода $\omega_{ra(j)}$, определенном выравниванием a.

Модель перевода (2)

$$P(a|\Pi_e, \Pi_r) = \frac{P(\Pi_e, a|\Pi_r)}{\sum_a P(\Pi_e, a|\Pi_r)}$$

Имея набор выравниваний с определенными вероятностями, мы можем подсчитать частоты каждой пары слов,

$$t(\omega_e|\omega_r) = \frac{counts(\omega_e|\omega_r)}{\sum_{\omega_e} counts(\omega_e|\omega_r)} = \frac{counts(\omega_e|\omega_r)}{total(\omega_r)};$$

Требуется оценить вероятности *лексического перевода* $t(\omega_e|\omega_r)$ Но чтобы сделать это нужно вычислить a, которой у нас нет.

Модель перевода (3)

Для оценки параметров $\longrightarrow$ EM-алгоритм (Витерби).

- инициализируем параметры модели (одинаковыми значениями, на первой итерации);
- оценим вероятности отсутствующей информации;
- оценим параметры модели на основании новой информации;
- перейдем к следующей итерации.

Базовый-алгоритм(Θ_e, Θ_r)

$\forall\, \omega_e \in \Pi_e \in \Theta_e$:
  $\forall \omega_r \in \Pi_r \in \Theta_r$:
    $t(\omega_e|\omega_r) \leftarrow u, u \in \mathbb{R}$;
▷ *Инициализируем таблицу $t(\omega_e|\omega_r)$ одинаковыми значениями.*
пока не сойдется :
  $\forall\, \omega_e \in \Pi_e \in \Theta_e$: ▷ *Инициализируем остальные таблицы.*
    $\forall \omega_r \in \Pi_r \in \Theta_r$:
      $counts(\omega_e|\omega_r) \leftarrow 0$; $total(\omega_r) \leftarrow 0$;
  $\forall\, \Pi_e, \Pi_r \in \Theta_e, \Theta_r$: ▷ *Вычисляем нормализациию.*
    $\forall\, \omega_e \in \Pi_e$:
      $stotal(\omega_e) \leftarrow 0$;
      $\forall\, \omega_r \in \Pi_r$:
        $stotal(\omega_e) \leftarrow stotal(\omega_e) + t(\omega_e|\omega_r)$;
    $\forall\, \omega_e \in \Pi_e$: ▷ *Собираем подсчеты.*
      $\forall\, \omega_r \in \Pi_r$:
        $$counts(\omega_e|\omega_r) \leftarrow counts(\omega_e|\omega_r) + \frac{t(\omega_e|\omega_r)}{stotal(\omega_e)};$$
        $$total(\omega_r) \leftarrow total(\omega_r) + \frac{t(\omega_e|\omega_r)}{stotal(\omega_e)};$$
  $\forall\, \omega_e \in \Theta_e$: ▷ *Оцениваем вероятность.*
    $\forall \omega_r \in \Theta_r$:
      $$t(\omega_e|\omega_r) \leftarrow \frac{counts(\omega_e|\omega_r)}{total(\omega_r)};$$

∃ Детали работы декодера (1)

- **В первом** режиме работы принимается исходный текст.
  - Последовательно разбивается на *n*-граммы.
    - Сначала наибольшего размера.
  - *n*-граммы ищутся в базе данных.
    - Если нашли, выбираем наиболее вероятную.
    - Если нет, берем *n*-грамму меньшего размера, слова (1-граммы) возвращаем как есть.
  - Вычисляем величину неопределенности.
- **Во втором** режиме работы на вход принимается:
  - исходный текст (ИТ);
  - переводной текст (ПТ) с предыдущей итерации
  - величина неопределенности (ВН).

∃ Детали работы декодера (2)

$$\text{BH} = 2^{-\left(\frac{1}{S_{\eta e}}\sum_{i=1}^{S_{\eta e}}\log_2 P(\eta_{ei}) + \frac{1}{S_{\omega e}}\sum_{j=1}^{S_{\omega e}}\log_2 P(\omega_{rj}|\omega_{ej})\right)}$$

- η_e — n-граммы найденные в созданном тексте;
- S_{η_e} — количество таких n-грамм;
- $P(\eta_e)$ — вероятность n-грамм согласно языковой модели (вычисляется как указано раннее);
- ω_e — n-граммы (слова) как результат перевода согласно модели перевода;
- S_{ω_e} — количество таких n-грамм (слов);
- $P(\omega_{rj}|\omega_{ej})$ — вероятность перевода фразы ω_{ej} на ω_{rj}.

BLEU — Bilingual Evaluation Understudy

$$\text{BLEU} = Bp \cdot e^{\left(\sum_{n=1}^{N} W_n \log(p_n)\right)}$$

$$Bp = \begin{cases} 1, & l_c > l_h; \\ e^{\left(1-\frac{l_h}{l_c}\right)}, & l_c \leqslant l_h. \end{cases} \quad \text{и} \quad p_n = \frac{\sum_{C \in S_c} \sum_{\eta_c \in C} \text{число}_{\text{среза}}(\eta_c)}{\sum_{C \in S_c} \sum_{\eta_c \in C} \text{число}(\eta_c)}$$

- S_c — множество кандидатов на перевод;
- C — кандидат на перевод;
- η_c — n-грамма кандидата на перевод;
- l_c — длинна кандидата перевода;
- l_h — длинна экспертного перевода (выполненного человеком);
- $W_n = \frac{1}{N}$ — вес;
- $N = 4$, n-граммность оценки.

Система	BLEU
ПРС-СМП (1)	0.243
ПРС-СМП (100)	0.209
Moses (IBM 3)	0.201
Moses (IBM 5)	0.173