① → 10

② → 11

③ → 14

④ → 27

⑤ → 28

⑥ Применение тезаурусов для расширения запросов.
Информационно-поисковый тезаурус (ИПТ) - нормативный словарь терминов предметной области, создаваемый для улучшения качества поиска в данной области.
Расширение запроса:

- для каждого термина t в запросе происходит расширение запроса словами близкими по смыслу.
- расширение → вставлять в вектор запроса с более низкими весами
  → вставлять в булевское выр-е

Используется в предметно-ориентированных системах.

(+) увеличение полноты поиска (−) • обычно снижает точность для многозначных слов
• сложность создания и обновления тезауруса

⑦ Метрики качества для систем автоматической классификации/рубрикации). Микро- и макро-усреднение.

P - точность, R - полнота

$P = \frac{TP}{TP+FP}$, $R = \frac{TP}{TP+FN}$

Комбинированная мера: $F1 = \frac{2PR}{P+R}$

Макроусреднение:
- посчитать F1 для каждого класса
- посчитать среднее

Микроусреднение:
- посчитать TP, FP, FN для каждого класса
- суммировать каждый показатель
- посчитать F1 для сумм

(8) → 44

(9) → 45

(10) → 46

(11) → 47

(12) Ф-ла подсчёта и особенности применения меры Mutual information

$MI = \log_2 \frac{f(a,b) \cdot N}{f(a) \cdot f(b)}$

N - размер корпуса
f - частота встречаемости

MI:
- завышает значимость редких словосочетаний, но при этом выявляются случайные словосочетания (опечатки)
- требует порог отсечения по частоте снизу
- обычно подбирается пороговое значение сверху

(13) Что такое термины? (опр., примеры)

Термин - слово или словосочетание, являющееся точным обозначением понятия какой-либо спец. области науки, техники, искусства и т.д.

Примеры: уравнение окружности, разностный метод, формальный параметр.

(14) Признаки для автоматического извлечения терминов из текстов.

1) Частотные признаки
- tf-idf
- модификация tf-idf с тестовой и контрастной коллекцией (idf берётся по контрастной коллекции)

2) Признаки, использующие контекстную информацию
- C-Value = $TF - \frac{\sum_{p \in P} TF(p)}{|P|}$, P - мн-во словосочетаний с данным словом.

Идея - объединение ~~[illegible]~~ частотности слов-кандидатов и информации о контекстных словах

3) Лингвистические признаки
- только прил. и сущ.
- сущ. в и.п.
- слова с заглавной буквы не в начале предл.

15 Автоматическое аннотирование. Виды автоматических аннотаций.

Автоматическое аннотирование – автоматическая технология, передающая в краткой форме основное содержание документа

Назначение: быстрое ознакомление с содержанием

Типы аннотаций:

1) абстракты vs. экстракты
   - абстракты → порождаются
   - экстракты → получаются извлечением фрагментов исх. текстов

2) по содержанию
   - индикативные (упор на само содержание)
   - информативные (упор на цифры, данные)
   - оценочные (упор на мнения)

3) по фокусу → общее содержание
   → в ответ на запрос

4) по структуре → связная аннотация
   → ключевые слова

5) по кол-ву документов

6) по используемому языку

16 Методы и признаки для отбора предложений в экстрактивном методе.

1) Частотные подходы

a) • метод SumBasic - наиболее частотные слова с большей вероятностью должны оказаться в аннотации

• на каждой итерации происходит расчёт вероятностей слов ⇒ предл. с max средней вероятностью слов

• пересчёт вероятностей для слов из отобранного предл. ⇒ снижение повторов

b) использование tf-idf
   - в исх. док.
   - в корпусе, из которого извлечён док-т

2) Подходы, основанные на методах опт.

a) • локальная оптимизация - оптимизация выбора следующего предложения

b) • глобальная оптимизация - оптимизация кек. ф-ции
   • ЛП

c) метод MMR

3) Графовые методы: вершины - предложения, дуги – сходство

4) Методы на основе машинного обучения

Характеристики для извлечения предлож.

- частотные слова текста
- учёт заголовка
- присутствие ключевых слов и конструкций
- позиция в тексте
- связность с предыдущим
- новизна информации

(17) Метод MMR

Итеративный метод - на каждой итерации прогр-ся ранжирование предложений - кандидатов

Q - запрос к системе

S - мн-во предложений-кандидатов

$s \in S$ - рассматриваемое предложение

E - мн-во выбранных предлож.

$$MMR = \arg\max_{s \in S} [\lambda Sim_1(s, Q) - (1-\lambda) \max_{s_j \in E} Sim_2(s, s_j)]$$

(18) Метрика Rouge

Идея - сравнение сгенерированного текста с эталонным составленным человеком

$$Rouge\text{-}N(A_i) = \frac{\sum_{M_{ij}} count(Ngram(A_i) \cap Ngram(M_{ij}))}{\sum_{M_{ij}} count(Ngram(M_{ij}))}$$

A_i - оцениваемая обзорная аннотация

M_{ij} - ручные аннотации i-го кластера

$Ngram(D)$ - мн-во всех n-грамм из лемм документа D.

(19) Метод пирамид для тестирования автоматических аннотаций

- Эксперты выделяют из "эталонных" аннотаций "информационные единицы" - SCU
- Каждая SCU получает вес, равный кол-ву "эталонных" аннотаций, где она встречалась
- Оценка - суммарный вес входящих SCU
- Итоговый рез = $\frac{\text{сумма весов SCU в авторских аннот.}}{\text{сумма весов SCU в экспертных аннот.}}$

(20) Исправление несловарных ошибок на основе применения правила Байеса.

x - неправильное слово

$\hat{w}$ - правильное:

$$\hat{w} = \arg\max_{w \in V} P(w|x) = \arg\max_{w \in V} \frac{P(x|w) P(w)}{P(x)} =$$

$$= \arg\max_{w \in V} P(x|w) P(w).$$

Порождается мн-во кандидатов.

$P(w) = \frac{C(w)}{T}$ ← кол-во вхождений w в корпус

↑ корпус

$P(x|w)$ - вероятность перехода/исправления (удаление, вставка, замена, перестановка).

$$P(x|w) = \begin{cases} \frac{del[w_{i-1}, w_i]}{count[w_{i-1}w_i]} & \text{- удаление} \\ \frac{ins[w_{i-1}, x_i]}{count[w_{i-1}]} & \text{- вставка} \\ \frac{sub[x_i, w_i]}{count[w_i]} & \text{- замена} \\ \frac{trans[w_i, w_{i+1}]}{count[w_i w_{i+1}]} & \text{- перест.} \end{cases}$$

$del[x,y]$ - кол-во xy написано как x

$ins[x,y]$ - кол-во x написано как xy

$sub[x,y]$ - кол-во y написано как x

$trans[x,y]$ - кол-во xy написано как yx

(21) Исправление ошибок перехода в другое словарное слово на основе применения метода Байеса.

Список кандидатов: само слово; слова на расстоянии 1-2; слова, близкие по звучанию.

Выбор лучшего кандидата - noisy channel:

- дано предложение $w_1, w_2, .., w_n$.
- мн-во кандидатов для каждого w_i:

$candidate(w_1) = \{w_1, w_1', w_1'' \dots\}$

$\vdots$

$candidate(w_n) = \{w_n, w_n', w_n'' \dots\}$

- выбрать посл-ть $W = argmax\, P(w)$

Вероятности пересчитываются по корпусу + channel model как для нечеловеческой ошибки + $P(w/w) \approx 0.9 .. 0.99$

(22) Учёт контекста при исправлении ошибок написания

- учёт условных вероятностей появления одного слова после другого:

$$P(w_1 .. w_n) = P(w_1) \cdot P(w_2|w_1) \dots P(w_n|w_{n-1})$$

- вероятности насчитываются на корпусе, нужно посчитать $P(w_i)$ - вероятность унограмм и $P(w_n|w_{n-1})$ - вероятность биграмм
- для унограмм $P(w_i) > 0$, но $P(w_i, w_{i-1})$ м.б. $= 0$

⇒ нужно сглаживание

(23) Методы тестирования В-О системы. Отличия от тестирования стандартной системы ИП.

Особенности ВО систем:

- нужно не только выдать документ, но и выдать фрагмент с ответом на вопрос
- длинные формулировки вопросов ⇒ многие

слова, учёт синтаксической структуры, в тексте могут быть синонимы

- тип вопроса и ответа должен совпадать

Методы оценки в TREC:

- Засчитываются первые три ответа системы
- Метрика MRR - за правильный ответ на 1-м месте - 1 балл, на втором - 0.5, на третьем - 0.25
- Вычисляется среднее по всему мн-ву вопросов

24 Позиционный индекс в поисковой системе. Зачем нужен, как обрабатывается.

Процесс индексирования в ПС:

- идентифицирует и сохраняет тексты для индексирования
- трансформирует документы в индексные термы
- берёт индексные термы и создаёт индексы для быстрого поиска

Создание индекса:

- собирается статистика по документам (частоты и позиции слов и других признаков) Эта инф. используется в ранжирующем алгоритме
- вычисление веса для индексного терма (e.g. tf×idf)
- инвертирование индекса - преобразование матрицы документ-терм в данные терм-документ ($t \to [docId]$)
- сжатие данных, распределённое хранение индекса - для быстрой обработки запросов

25 PageRank (зачем нужен, как вычисляется)

Пользователь начинает со случайной страницы, на каждом шаге переходит на следующую с равной вероятностью => в пределе каждая страница получает рейтинг посещений ~ вес страницы.

С тупиковых страниц - переход на случайную.

$$PR(A) = d + (d-1)\left(\frac{PR(T_1)}{C(T_1)} \dots \frac{PR(T_n)}{C(T_n)}\right)$$

$PR(T_n)$ - исходная значимость

$C(T_n)$ - кол-во исходящих ссылок со стр.

d - значимость страницы без входящих ссылок (коэф. телепортации)

Матем. основа - марковские цепи из n состояний. P - матрица переходов nxn. PageRank - вектор стационарного состояния ($a = aP$ или считать итеративно $xP, xP^2, \dots$, x - начальный вектор).

24 Позиционный индекс

Используется для обработки фразовых запросов

Хранятся позиции терма в каждом документе

Для сопоставления используется алгоритм merge.

Позиционный индекс стандартно используется для обработки запросов на близость и фразовых запросов.

26 Методы приблизительного вычисления сходства документов в реальных поисковых системах.

1) Неточная векторная модель:

- найти A кандидатов $K < |A| << N$; A не обязательно содержит все k лучших документов, но содержит многие из них
- выдать K лучших из A
- Методы: рассмотреть только слова с высоким idf; рассмотреть только документы с большим кол-вом терминов запроса

2) Champion lists

- заранее для каждого терма t вычисляется r документов с max весом t.
- r выбирается во время индексирования и может зависеть от t.
- во время запроса считаются только веса документов, входящих хотя бы в один топ-лист одного из термов запроса
- выбираются K лучших документов из топ-листов

3) Учёт качества документа

- авторитетность - кач-во документа, не зависящее от запроса
- примеры факторов: ср. Wiki, статьи из СМИ научные статьи с большим кол-вом цитирований
- $g(d)$ - вес кач-ва
- $net\text{-}score = \alpha g(d) + (1-\alpha) \times cos(q, d)$
- храним список лучших документов для слова по net-score
- ищем лучшие k результатов только из этого списка

27 Обработка фразовых запросов и вопросов с указанием близости слов

Для фразовых запросов недостаточно хранить только индекс терм-документ

1) Биграммный индекс: строится на парах слов из запроса

• длинные запросы нарезаются на пары

− могут встретиться док, которые содержат пары слов, но не фразу;
сверхбольшой индекс;
проблемы со случайным вхождением слов в биграмму

2) Позиционный индекс – см. №24

Обработка запросов с операторами близости:

• может использоваться только позиционный индекс

• возможно комбинирование позиционного и биграммного

Разборщик запросов – запрос исполняется как несколько запросов к системе, пока не будет набрано К лучших документов

(28) Как учитывается при поиске различные зоны документа

Документ имеет много частей, имеющих свой смысл: автор, заголовок, дата публикации, язык, формат – метаданные

Создаются отдельные индексы по зонам метаданных – параметрический индекс

Многоуровневые списки:

• разбиение индексов по уровням

• инвертированный список разбивается на уровни снижающейся важности

• при поиске используются индексы большей важности

(29) Какие факторы помимо веса tf×idf учитываются в поисковых моделях, как создаются многофакторные модели и т.п.

Факторы для агрегирования весов:

• скалярное пр-е

• близость слов запроса

• ссылки

• соотв-е тематике запроса и док-та

Комплексная ф-я релевантности:

• частота термов

• совместная встречаемость термов

• зоны док-та

• неполное вхождение слов в запрос

• особая обработка длинных документов

$$W = k_f W_f + k_p W_p + k_{ps} W_{ps}$$

$k_f W_f$ – частотность термов по tf·idf

$k_p W_p$ – встречаемость пар соседних слов запроса

$k_{ps} W_{ps}$ – вес наилучшего пассажа в док-те

30 Методы поискового спама

- повторы терминов
- "мета-теги" с избыточным повторением
- клоакинг - информация, выдаваемая краулеру и пользователю, различается
- дорвеи - страницы оптимизируются под заданное ключевое слово, которое перенаправляет на нужную страницу
- ссылочный спам, линкофермы (покупка ссылок)
- порождение искусственных текстов

31 Методы противодействия поисковому спаму

- сигналы качества - голоса авторов, голоса пользователей
- ограничения на мета-слова
- анализ ссылок
- использование машинного обучения
- редакторская политика: чёрные листы, прослушр частотных запросов, жалобы пользователей, шаблоны для обнаружения подозрительных сайтов

? 32 Байесовский классификатор в задаче обнаружения поискового спама

Необходима коллекция извесного спама - используется для обучения.

Классификатор определяет вероятность того, что данная страница/сайт содержит спам.

33 Методы порождения неестественных текстов и зачем это нужно в поисковой оптимизации?

1) Цепи Маркова (ищется в исходном тексте каждое следующее слово)
2) Метод фокуса внимания
3) Метод с использованием словарей

При поисковой оптимизации сайтов исп-ся для генерации названий, описаний и содержимого целых сайтов с большим кол-вом страниц и "уникальным контентом"

34 Алгоритм HITS

В ответ на ~~запрос~~ запрос вместо упор. мн-ва страниц выдаётся 2 мн-ва взаимосвязанных страниц:

- хабы - хорошие списки ссылок по теме
- авторитеты - часто упоминаются в хабах

Хорошая хаб-страница указывает на многие авторитетные страницы

Хорошая авторитетная страница указывается большим кол-вом хороших хабов

Для каждой страницы рекурсивно вычисляется её значимость как посредника и авторитета

Построение базового набора страниц:

1) по запросу получаем страницы, содержащие слова запроса - это корневой набор

2) добавляем ∀ страницу, которая указывает на стр. из корневого набора или на которую есть из него ссылка
⇒ получим базовый набор

Разделение хабов и авторитетов:

1) Инициализация hub score $h(x)$ и authority score $a(x)$

2) Итеративно пересчитываем: $\forall x$

$$h(x) \leftarrow \sum_{x \to y} a(y)$$

$$a(x) \leftarrow \sum_{y \to x} h(y)$$

$a(x)$ вычисляется как сумма значений оценок посредственных страниц, которые на неё указывают

$h(x)$ вычисляется как Σ значений оценок авторитетности страниц, на кот. она указывает

Значения в итоге сходятся.

35 Метод шинглов для определения дубликатов документов

Шинглы - порядковые n-граммы

1) Канонизация текстов - очищение текстов от слов и символов, которые не должны участвовать в сравнении; нормализация

2) Разбиение на шинглы вычисляется

3) Вычисление контрольных сумм шинглов через 84 независимые статические хэш-функции.

4) Случайная выборка из каждого набора хэшей

5) Сравнение полученных массивов на соответствие

36 Особенности использования кликов пользователя в качестве фидбека

Отсутствие кликов - не релевантно/не видно

Неравномерность позиций относит-но кликов:

- более высокие позиции получают больше кликов
- справедливо, если переставить выдачу
⇒ клики информативны, но смещены

Гипотеза о наблюдении:

- документ должен быть прочитан перед кликом; условная вероятность клика после прочтения зависит от релевантности док-та
- вероятность клика - ~~гл~~ глобальный компонент (вер-ть увидеть, зависит от позиции), локальный компонент (зависит от пары <запрос, документ>)

37 Классификация запросов по цели. Особенности обработки разных типов запросов.

1) Информационные запросы - пользователю важен конечный результат

2) Навигационные запросы - для поиска конкретного адреса веб-ресурса

3) Транзакционные запросы - для приобретения конкретных товаров и услуг

Имея представление о классах запросов, можно глубже анализировать выдачу, строить гипотезы для изучения алгоритмов ранжирования, совершенствовать и оптимизировать сайт.