|  |  |  |
| --- | --- | --- |
| ***1.Понятие информационной технологии.***  Блок рандомизации  Ключ Адрес  Ключ1 Адрес1  Ключ2 Адрес2  Ключ3 Адрес3  Текст1  Адрес11  Текст2  Адрес21  Адрес22  Текст3  Ключ Адрес  Ключ1 Адрес1 Адрес11  Ключ2 Адрес2 Адрес21  Ключ3 Адрес3  Текст1  Текст2  ИТ – средства, методы и системы сбора, передачи, обработки и представления информации пользователю.  Существуют 4 этапа эволюции:   * Появление речи; * Появление письменности; * Появление средств ВТ (Современные и тд.). * В современном ИТ выделяют 3 составляющие: * Аппаратное обеспечение (средства ВТ и оргтехники – hardware); * Программное обеспечение (прикладное и системное ПО, методическое и информационное обеспечение – software); * Организационное обеспечение (включая человека в системы ИТ, взаимодействие человека с этими системами, системное использование технических и программных средств – orgware)   ИТ = новые ИТ = современные ИТ.  Новые ИТ – современная ИТ технология, использующая развитый (интеллектуальный) интерфейс с конечным пользователем.  ИТ как прикладная наука, изучает фундаментальные соотношения в больших информационных системах.  ИТ как практика – интеллектуальная деятельность по проектированию и созданию конкретных технологий обработки данных.  В ИТ выделяют 3 составляющие:   1. Базовые ИТ. Обеспечивают решение отдельных компонентов в той или иной задаче, служат для создания прикладных ИТ. Например: технологии программирования, СУБД, системы распознавания изображения и тд. 2. Прикладные ИТ. Формируются на основе базовых ИТ, предназначены для полной информатизации объекта. Например: САПРы, АСУП, геоинформационные системы. 3. Обеспечивающие ИТ. Обеспечивают реализацию базовых и прикладных ИТ. На рынке представлены их отдельные компоненты. Например: современная микроэлектронная база средств ВТ, перспективные системы и комплексы (оптические и нейрокомпьютеры, транспьютеры).   ***3.Основные методы организации текстовых файлов.***   1. Цепочечные файлы.   К самой БД добавляется справочник, который имеет следующую структуру:  Ключ – значимое слово, характеризующее тот или иной документ. Рядом пишется адресная ссылка на тот текстовый файл, который имеет данный ключ в качестве значимого термина. И к этой подстроке добавляются собственно текстовые файлы.  Цепочечная модель: сколько индексных терминов в тексте выделено столько и должно быть ссылок.  Преимущества:   * Максимальная длина поиска определяется самой длинной цепочкой; * Новые записи (тексты) можно ставить в начало цепи, что упрощает её корректировку. * Недостатки: * Цепи могут быть длинными, если некоторые ключи используются довольно часто; * Необходимость выделения памяти для хранения адресных ссылок в самих текстах; * Если справочник очень велик, он значительно усложняет работу с текстами и требует организации дополнительного доступа к себе самому.   ***Вопрос 5(окончание).***  термину k. Если Sk уменьшается, то k либо вообще не рассматривается как возможный индекс, либо ему присваивается отрицательный вес.  1.Параметры, основанные на динамической эффективности. Всем терминам первоначально присваиваются одинаковый вес, затем пользователь формирует запрос, и выдаются документы и пользователь определяет релевантность, система сама уменьшает или увеличивает вес документа, в соответствии с потребностями пользователя, т.е. предусматривается некоторая программа обучения системы.  Мы рассмотрели статистические подходы (СП). Помимо СП используются такие подходы, которые предусматривают местоположение термина в тексте.  Подходы:  1. В индексационные термины включаются те, которые встречаются в названиях документов, названиях глав, разделов и т.д.  2. Составляются списки значимых для некоторой предметной области слов. Т.е. составляется глоссарий по некоторой предметной области.   1. Методы увеличения полноты. Часто бывает необходимо выдать наибольшее число релевантных документов из массива. В этом случае необходимо к используемым индексационным терминам добавить дополнительные, чтобы расширить область поиска. 2. 1-й подход к решению этой задачи: использование терминов заместителей из словаря синонимов, который называют тезариусом, в котором термины сгруппированы в классы.   2) Метод ассоциативного индексирования. Основан на использовании матрицы ассоциируемости терминов, которая задаёт для каждой пары терминов показатель ассоциируемости. Абсолютная запись этого ПА между терминами j и k:  f ( j ; k ) = сумм ( i=1 – n ) fij \* fjk – частота совместного использования f ( j ; k ) = сумм ( i=1 – n ) fij \* fjk / (сумм ( i=1 – n ) fij ^ 2+ сумм ( i=1 – n ) fjk ^ 2 – сумм ( i=1 – n ) fij \* fjk - для расчёта относительного значения этого показателя. fij,k – частота появления термина j или k в i – м документе. 0 <= f ( j ; k ) <= 1. Если f ( j ; k ) = 0, то термины совсем не ассоциируются, если f ( j ; k ) = 1, то полностью ассоциируемы.  ***6.Использование частотных мер в индексировании.***  Частотный метод – по каждому термину, входящему в документ подсчитывается частота вхождения терминов в документ fik, i – номер документа, k – термин. Эта частота абсолютная. Затем документы упорядочиваются в соответствии с возрастанием или убыванием частоты.  Если термин имеет большую частоту, то это, скорее всего общеупотребительный термин, не раскрывающий конкретную предметную область (будет много документов).  Если термин имеет малую частоту, то он существенно отражает содержание, даже если его включить в дескрипторы (ключевые слова), то он , скорее всего будет использоваться в холостую. Поэтому эти 2 простейших документа исключают из списка.  Терминам с большей частотой присваивают меньший вес, с меньшей частотой – больший вес. | 1. Инвертированные файлы.   Получаются из цепочечных файлов, когда в справочник включаются адресные ссылки на все тексты, имеющие соответствующий ключ в качестве индексационного термина.  Недостаток: переменное число адресов в справочнике.  Достоинство: быстрый поиск релевантных документов, так как их адреса находятся сразу в справочнике, обработку которого можно организовать в оперативной памяти.   1. Рассредоточенные файлы.   Весь массив документов разбивается на группы файлов, ключевые термины которых связаны некоторым математическим соотношением. Тогда поиск в справочнике заменяется вычислительной процедурой, которая называется хешированием, рандомизацией или перемешиванием. Здесь нет справочника, а существует вычислительная процедура, т.е. блок, называемый блоком рандомизации, который по ключу (поисковому термину) на основании вычислительной процедуры определяет адрес, по которому находится текст.  Ключ адрес этот участок  {ключ} памяти  называется  бакетом  В этой области памяти находится несколько текстов, каждый из которых характеризуется по своему в векторе документов. Т.е. адрес получается по вычислительной процедуре.  Преимущества:   * Быстрый вычисляемый доступ; * Из-за отсутствия справочника экономится память.   Недостатки:   * Сложность при выборе метода хеширования; * Применяется для коротких векторов запросов, когда в поиске участвует немного слов; * Изменения векторов документов порождает сложность в ведении файлов.  1. Кластерные файлы.   Документы разбиваются на родственные группы, которые называют кластерами или классами. Каждый класс описывается центроидом (профилем) и вектор запроса прежде всего сравнивается с центроидами класса.  Преимущества:   * Возможен быстрый поиск, т.к. число классов, как правило, невелико; * Возможно интерактивное сужение (расширение) поиска за счёт исключения или добавления дополнительных кластеров.   Недостатки:   * Необходимость формировать кластеры; * Необходимость введения файла центроидов; * Дополнительный расход памяти для файла центроидов или профилей.     ***4. Понятие центроида кластера.***  Множество терминов составляющих векторов кластера называются центроидом или репрезентативным кластерным профилем. Т.о. каждый кластер характеризуется центроидным вектором, который представляет собой множество пар: {(tik , wik)}, где tik – множество терминов описывающих i-й кластер, wik – множество весов.  Вес – число, определяющее значимость данного термина для раскрытия содержимого документа.  ***7.Расчет соотношения “сигнал-шум” при индексировании.***  Использование соотношения “ сигнал – шум “. Здесь исключается ещё одна частота: суммарная или общая частота появления термина k в наборе из n документов и рассчитывается:  Fk = сумма (i=1 – n) fik  Шум k –го символа рассчитывается:  Nk = сумма(i=1 – n) fik / Fk \* log (Fk / fik)  Сигнал k – го символа:  Sk = log Fk – Nk  Шум является максимальным, если термин имеет равномерное распределение в n документах. Шум является минимальным и равномерным, когда термин имеет неравномерное распределение, например, когда он встречается только в одном документе, с частотой Fk, тогда:  Nk = сумма (i=1 – n) fik / Fk \* log Fk / fik = 0, в этом случае сигнал имеет максимальное значение:  Sk = log Fk – Nk = log Fk  С учётом этих параметров, для определения веса используется отношение сигнала к шуму k –го термина: Sk / Nk. Чем больше это отношение, тем больший вес назначается. Строится однозначная таблица.  ***8.Использование распределения частоты термина при индексировании.***  Использование распределения частоты термина (уклонения).  Уклонение рассчитывается:  U = (сумм (fik – fk)) / (n-1)  fk – средняя частота термина k в наборе из n документов.  fk = Fk / n  Для оценки веса термина используется не уклонение, а формула Fk\* U/ fk  Чем больше это отношение, тем больший вес назначается термину.  ***9.Использование при индексировании параметров, основанных на способности термина различать документы набора.***  Исходные данные – набор из n документов и множество S коэффициентов подобия всех пар документов из множества n: { S ( Di , Dj ) }. Эти коэффициенты подобия рассчитываются на основании векторов документов. Способ расчета разный, а принцип: S ( Di , Dj ) = 1, если вектора идентичны.  S ( Di , Dj ) = 0 , если в векторах нет ни одного общего документа.  По S рассчитывают средний коэффициент подобия: S = C \* сумм (i= 1 – n) S ( Di , Dj ), С – коэффициент усреднения, может быть любым, в частности C = 1 / n.  Далее из векторов документов удаляют некоторый k – й термин и рассчитывают средний коэффициент по парного подобия, но с удалённым k –м термином: Sk( т.е. в векторах документа не участвуют веса k –го термина). Если Sk возрастает относительно S, то термину k присваивается положительный вес. Чем больше эта разница, тем больший вес присваивается термину k. Если Sk уменьшается, то k либо вообще не рассматривается как возможный индекс, либо ему присваивается отрицательный вес.  ***10.Динамическая информативность как метод индексирования.***  Всем терминам первоначально присваиваются одинаковый вес, затем пользователь формирует запрос, и выдаются документы и пользователь определяет релевантность, система сама уменьшает или увеличивает вес документа, в соответствии с потребностями пользователя, т.е. предусматривается некоторая программа обучения системы. | ***5 Назначение и основные методы индексации.***  Задача создания вектора документа называется индексированием.  Методы автоматического индексирования. Задачи этих методов – построить векторы документов {(tik , wik)}. Исходные данные – массив документов. Нужно выделить те термины, которые раскрывают текст документа tik и присвоить вес wik.  Методы:  1. Частотный метод – по каждому термину, входящему в документ подсчитывается частота вхождения терминов в документ fik, i – номер документа, k – термин. Эта частота абсолютная. Затем документы упорядочиваются в соответствии с возрастанием или убыванием частоты.  Если термин имеет большую частоту, то это, скорее всего общеупотребительный термин, не раскрывающий конкретную предметную область (будет много документов).  Если термин имеет малую частоту, то он существенно отражает содержание, даже если его включить в дескрипторы (ключевые слова), то он , скорее всего будет использоваться в холостую. Поэтому эти 2 простейших документа исключают из списка.  Терминам с большей частотой присваивают меньший вес, с меньшей частотой – больший вес.  2. Использование соотношения “ сигнал – шум “. Здесь исключается ещё одна частота: суммарная или общая частота появления термина k в наборе из n документов и рассчитывается:  Fk = сумма (i=1 – n) fik  Шум k –го символа рассчитывается:  Nk = сумма(i=1 – n) fik / Fk \* log (Fk / fik)  Сигнал k – го символа:  Sk = log Fk – Nk  Шум является максимальным, если термин имеет равномерное распределение в n документах. Шум является минимальным и равномерным, когда термин имеет неравномерное распределение, например, когда он встречается только в одном документе, с частотой Fk, тогда:  Nk = сумма (i=1 – n) fik / Fk \* log Fk / fik = 0, в этом случае сигнал имеет максимальное значение:  Sk = log Fk – Nk = log Fk  С учётом этих параметров, для определения веса используется отношение сигнала к шуму k –го термина:  Sk / Nk. Чем больше это отношение, тем больший вес  Назначается. Строится однозначная таблица.  1.Использование распределения частоты термина (уклонения).  Уклонение рассчитывается:  U = (сумм (fik – fk)) / (n-1)  Fk – средняя частота термина k в наборе из n документов.  Fk = Fk / n  Для оценки веса термина используется не уклонение, а формула Fk\* U/ fk  Чем больше это отношение, тем больший вес назначается термину.  2.Параметры, основанные на способности термина различать документы набора. Исходные данные – набор из n документов и множество S коэффициентов подобия всех пар документов из множества n: { S ( Di , Dj ) }. Эти коэффициенты подобия рассчитываются на основании векторов документов. Способ расчета разный, а принцип: S ( Di , Dj ) = 1, если вектора идентичны.  S ( Di , Dj ) = 0 , если в векторах нет ни одного общего документа.  По S рассчитывают средний коэффициент подобия:S = C \* сумм (i= 1 – n) S ( Di , Dj ), С – коэффициент усреднения, может быть любым, в частности C = 1 / n.  Далее из векторов документов удаляют некоторый k – й термин и рассчитывают средний коэффициент по парного подобия, но с удалённым k –м термином: Sk( т.е. в векторах документа не участвуют веса k –го термина). Если Sk возрастает относительно S, то термину k присваивается положительный вес. Чем больше эта разница, тем больший вес присваивается  ***11.Методы индексирования, основанные на положении термина в тексте.***  Подходы:  1.В индексационные термины включаются те, которые встречаются в названиях документов, названиях глав, разделов и т.д.  2.Составляются списки значимых для некоторой предметной области слов. Т.е. составляется глоссарий по некоторой предметной области.  3.Методы увеличения полноты. Часто бывает необходимо выдать наибольшее число релевантных документов из массива. В этом случае необходимо к используемым индексационным терминам добавить дополнительные, чтобы расширить область поиска.  1)1-й подход к решению этой задачи: использование терминов заместителей из словаря синонимов, который называют тезариусом, в котором термины сгруппированы в классы.  2)Метод ассоциативного индексирования. Основан на использовании матрицы ассоциируемости терминов, которая задаёт для каждой пары терминов показатель ассоциируемости. Абсолютная запись этого ПА между терминами j и k:  f ( j ; k ) = сумм ( i=1 – n ) fij \* fjk – частота совместного использования f ( j ; k ) = сумм ( i=1 – n ) fij \* fjk / (сумм ( i=1 – n ) fij ^ 2+ сумм ( i=1 – n ) fjk ^ 2 - сумм ( i=1 – n ) fij \* fjk - для расчёта относительного значения этого показателя. fij,k – частота появления термина j или k в i – м документе. 0 <= f ( j ; k ) <= 1. Если f ( j ; k ) = 0, то термины совсем не ассоциируются, если f ( j ; k ) = 1, то полностью ассоциируемы.  ***12.Постановка задачи увеличения полноты при поиске в текстовой базе данных и основные методы ее решения.***  Методы увеличения полноты. Часто бывает необходимо выдать наибольшее число релевантных документов из массива. В этом случае необходимо к используемым индексационным терминам добавить дополнительные, чтобы расширить область поиска.  1)1-й подход к решению этой задачи: использование терминов заместителей из словаря синонимов, который называют тезариусом, в котором термины сгруппированы в классы.  2)Метод ассоциативного индексирования. Основан на использовании матрицы ассоциируемости терминов, которая задаёт для каждой пары терминов показатель ассоциируемости. Абсолютная запись этого ПА между терминами j и k:  f ( j ; k ) = сумм ( i=1 – n ) fij \* fjk – частота совместного использования f ( j ; k ) = сумм ( i=1 – n ) fij \* fjk / (сумм ( i=1 – n ) fij ^ 2+ сумм ( i=1 – n ) fjk ^ 2 – сумм ( i=1 – n ) fij \* fjk - для расчёта относительного значения этого показателя. fij,k – частота появления термина j или k в i – м документе. 0 <= f ( j ; k ) <= 1.  Если f ( j ; k ) = 0, то термины совсем не ассоциируются, если f ( j ; k ) = 1, то полностью ассоциируемы.  Второй способ: используются матрицы для расширения поиска: вводится некоторое пороговое значение коэффициента ассоциируемости (СКА), выше которого коэффициенты приравниваются к единице, а ниже к 0. |
| ***13.Метод ассоциативного индексирования в задаче увеличения полноты поиска.***  Основан на использовании матрицы ассоциируемости терминов, которая задаёт для каждой пары терминов показатель ассоциируемости. Абсолютная запись этого ПА между терминами j и k:  f ( j ; k ) = сумм ( i=1 – n ) fij \* fjk – частота совместного использования f ( j ; k ) = сумм ( i=1 – n ) fij \* fjk / (сумм ( i=1 – n ) fij ^ 2+ сумм ( i=1 – n ) fjk ^ 2 - сумм ( i=1 – n ) fij \* fjk - для расчёта относительного значения этого показателя. fij,k – частота появления термина j или k в i – м документе. 0 <= f ( j ; k ) <= 1.  Если f ( j ; k ) = 0, то термины совсем не ассоциируются, если f ( j ; k ) = 1, то полностью ассоциируемы.  Второй способ: используются матрицы для расширения поиска: вводится некоторое пороговое значение коэффициента ассоциируемости (СКА), выше которого коэффициенты приравниваются к единице, а ниже к 0.  ***14.Метод вероятностного индексирования в задаче увеличения полноты поиска.***  Суть: наличие в документе некоторых терминов Т1, Т2, …, Тi позволяет с некоторой вероятностью Р отнести эти документы к классу документов Ск и присвоить вектору документов идентификатор этого класса, т.е. дополнительный термин. Причём указанная вероятность Р для этого “приписывания” должна быть больше некоторого порогового значения.  Вероятность Р записывается: Р(Т1, Т2, …, Тi , Ск) – вероятность того, что при наличии терминов Тi, документ будет принадлежать классу Ск. Р(Т1, Т2, Тi, Ск) = а \* р(Ск) \* р(Т1, Ск) \* (Т2, Ск) \* … \* (Тi, Ск)  Коэффициент а подбирается таким образом, чтобы выполнялось условие: сумма(к=1, m) Р(Т1, Т2, …, Тi , Ск) = 1 – т.е. чтобы выполнялась полная группа событий. Документ, содержащий термин Т1, Т2, …, Тi обязательно должен принадлежать одному из классов Ск.  m – число классов документов нашего массива.  Р(Ск) – вероятность класса Ск. Эта вероятность рассчитывается как частота, в числителе – число документов, находящихся в классе Ск, в знаменателе – общее число документов во всех m классах.  Р(Тj, Ск) – дробь, в числителе – общее число появления термина Тj в документах класса Ск, в знаменателе – общее число появления всех терминов в документах класса Ск.  ***15.Постановка задачи улучшения точности поиска в текстовой базе данных и основные методы ее решения.***  Задача - как можно точнее получать нужные документы.  2 способа:  а). Использование наиболее узких терминов.  б). Использование словосочетаний для индексирования документов. Для определения словосочетаний используются статистический и лингвистический подходы.  Статистический подход (СП):  В соответствии со СП словосочетание – такая комбинация терминов, частота совместного появления которых в массивах документов велика относительно частот появления отдельных терминов этого словосочетания. Связность терминов определяется коэффициентом связности:  Сik = Fkj / ( Fk \* Fj ) \* N – это связность 2-х терминов, хотя может быть и больше (до 4-х).  Fk , Fj – частные частоты терминов k и j. | ***Вопрос 15(окончание).***  Fkj – частота совместного появления терминов.  N – число слов в массиве.  После расчёта этих значений и коэффициента связности (КС) в словосочетании отбирают такие термины, для которых Сjк и Fkj больше порогового значения, которое устанавливается эмпирически. Пороговые значения: Сjк >= 20 и Fkj >= 3.  Если эти характеристики для термов, включённых в словосочетание поддерживаются, то получаются хорошие выборки.  Недостаток:   * не учитывается порядок слов в словосочетаниях; * метод позволяет считать идентичными даже словосочетания с одинаковым порядком следования термина.   Лингвистические методы – используют упрощённые синтаксические разборы предложений, причём, как правило, предложений из заголовков текстов.  Алгоритм анализа упрощённых фраз:   1. Образуются предводительные словосочетания путём проставления скобок перед предлогами, числительными, неопределёнными местоимениями и т.д. 2. Устанавливаются связи справа и/или слева от слов, выделенных в первом пункте между различными структурами. 3. Из структуры исключаются количественные числительные, вспомогательные глаголы, местоимения и т.д. Остаются лишь индексационные словосочетания. В результате должны остаться связи или комбинации вида: сущ. – сущ. (прил. - прил.).   Пример: (Some investigations)(in computer science)(which can lead)(to the creation)(of artificial intelligence). В результате имеем:  Computer science - >investigations -> artificial intelligence -> creation.  ***16.Статистический метод образования словосочетаний в задаче улучшения точности поиска в текстовой базе данных.***  В соответствии со СП словосочетание – такая комбинация терминов, частота совместного появления которых в массивах документов велика относительно частот появления отдельных терминов этого словосочетания. Связность терминов определяется коэффициентом связности:  Сik = Fkj / ( Fk \* Fj ) \* N – это связность 2-х терминов, хотя может быть и больше (до 4-х).  Fk , Fj – частные частоты терминов k и j.  Fkj – частота совместного появления терминов.  N – число слов в массиве.  После расчёта этих значений и коэффициента связности (КС) в словосочетании отбирают такие термины, для которых Сjк и Fkj больше порогового значения, которое устанавливается эмпирически. Пороговые значения: Сjк >= 20 и Fkj >= 3.  Если эти характеристики для термов, включённых в словосочетание поддерживаются, то получаются хорошие выборки.  Недостаток:   * не учитывается порядок слов в словосочетаниях; * метод позволяет считать идентичными даже словосочетания с одинаковым порядком следования термина. | ***17.Лингвистический метод образования словосочетаний в задаче улучшения точности поиска в текстовой базе данных.***  Лингвистические методы – используют упрощённые синтаксические разборы предложений, причём, как правило, предложений из заголовков текстов.  Алгоритм анализа упрощённых фраз:   1. Образуются предварительные словосочетания путём проставления скобок перед предлогами, числительными, неопределёнными местоимениями и т.д. 2. Устанавливаются связи справа и/или слева от слов, выделенных в первом пункте между различными структурами. 3. Из структуры исключаются количественные числительные, вспомогательные глаголы, местоимения и т.д. Остаются лишь индексационные словосочетания. В результате должны остаться связи или комбинации вида: сущ. – сущ. (прил. - прил.).   Пример: (Some investigations)(in computer science)(which can lead)(to the creation)(of artificial intelligence). В результате имеем:  Computer science - >investigations -> artificial intelligence -> creation.  ***18.Задача автоматического реферирования текстов и методы ее решения.***  Задача создания рефератов – задача выявления списка документов и краткое его представление.  Исходные данные:   1. массив исходных данных; 2. готовые вектора документов (т.е. уже должна быть решена задача создания векторов {tik , wik}).   Методы для автореферирования:   1. Расчётный – определяются веса словосочетаний, содержащих 2 значимых термина из вектора документа.   w = 1 / 2t \* wi1 \* wi2 – вес словосочетания из 2-х терминов .  wi1 и w i2 - веса 1-го и 2-го термина из вектора.  t – количество слов в тексте между терминами ti1 и t i2, которые не являются значимыми.  Далее по тексту определяется значимое предложение. Это такое предложение, которое содержит большое число значимых групп. После расчёта значимости предложений, они упорядочиваются и для реферирования выбираются наиболее значимые. Далее наиболее значимые предложения упорядочиваются так, как они шли в тексте, чтобы не потерять логику.   1. Позиционный метод. Включает в себя следующие не альтернативные шаги:   Наиболее значимые предложения, которые либо начинают, либо заканчивают абзац или раздел.  Исключаются вопросительные предложения, несмотря на их положение в абзаце.  К значимым относятся предложения, содержащие слова – подсказки. Например: “ данная (слово-подсказка) работа выполнена по такому – то плану и т.д.”  Из значимых исключаются те предложения, в которых есть ссылки на рисунки, таблицы, цитаты и т.д. |
| ***19.Цепочечные текстовые файлы.***  Ключ Адрес  Ключ1 Адрес1 Адрес11  Ключ2 Адрес2 Адрес21  Ключ3 Адрес3  Текст1  Текст2  Блок рандомизации  Ключ Адрес  Ключ1 Адрес1  Ключ2 Адрес2  Ключ3 Адрес3  Текст1  Адрес11  Текст2  Адрес21  Адрес22  Текст3  К самой БД добавляется справочник, который имеет следующую структуру:  Ключ – значимое слово, характеризующее тот или иной документ. Рядом пишется адресная ссылка на тот текстовый файл, который имеет данный ключ в качестве значимого термина. И к этой подстроке добавляются собственно текстовые файлы.  Цепочечная модель: сколько индексных терминов в тексте выделено столько и должно быть ссылок.  Преимущества:   * Максимальная длина поиска определяется самой длинной цепочкой; * Новые записи (тексты) можно ставить в начало цепи, что упрощает её корректировку. * Недостатки: * Цепи могут быть длинными, если некоторые ключи используются довольно часто; * Необходимость выделения памяти для хранения адресных ссылок в самих текстах; * Если справочник очень велик, он значительно усложняет работу с текстами и требует организации дополнительного доступа к себе самому.   ***20.Инвертированные текстовые файлы.***  Получаются из цепочечных файлов, когда в справочник включаются адресные ссылки на все тексты, имеющие соответствующий ключ в качестве индексационного термина.  Недостаток: переменное число адресов в справочнике.  Достоинство: быстрый поиск релевантных документов, так как их адреса находятся сразу в справочнике, обработку которого можно организовать в оперативной памяти.  ***21.Рассредоточенные текстовые файлы.***  Весь массив документов разбивается на группы файлов, ключевые термины которых связаны некоторым математическим соотношением. Тогда поиск в справочнике заменяется вычислительной процедурой, которая называется хешированием, рандомизацией или перемешиванием.  Здесь нет справочника, а существует вычислительная процедура, т.е. блок, называемый блоком рандомизации, который по ключу (поисковому термину) на основании вычислительной процедуры определяет адрес, по которому находится текст.  Ключ адрес этот участок  {ключ} памяти  называется  бакетом  В этой области памяти находится несколько текстов, каждый из которых характеризуется по своему в векторе документов. Т.е. адрес получается по вычислительной процедуре.  Преимущества:   * Быстрый вычисляемый доступ; * Из-за отсутствия справочника экономится память.   Недостатки:   * Сложность при выборе метода хеширования; * Применяется для коротких векторов запросов, когда в поиске участвует немного слов; * Изменения векторов документов порождает сложность в ведении файлов.   ***Вопрос 27(окончание).***  **4. Коррекция кластеров сверху вниз.**  В начале строятся один или несколько очень больших кластеров, которые затем разбиваются на более мелкие.  *Способы выбора исходных классов:*   * В качестве центров классов используются случайные документы; * Классом с именем i можно считать множество документов, в векторах которых находится термин i; * В качестве исходных классов принимаются все документы, признанные релевантными некоторому запросу по результатам предыдущих поисковых операций.   *Процесс коррекции кластеров:*   * Вычисляется КП между каждым документом и каждым центроидом кластера; * Кластеры переопределяются путём отнесения документов к тем из них, по отношению к которым, они имеют наибольшее подобие; * Формируются центроиды новых кластеров.   Эти 3 шага выполняются до тех пор, пока:   * Будет необходимость в изменениях; * Чтобы процесс не был бесконечным, он выполняется в заданное число итераций.   **5.Однократная кластеризация.**  Документы рассматриваются в произвольном порядке и каждый документ либо относится к существующему классу, если КП достаточен, либо образует новый кластер.  “+”: каждый документ обрабатывается только 1 раз, => требует мало времени.  “-”: состав и структура классов существенно зависит от порядка рассмотрения документов.  ***28.Нахождение КЛИК.***  Клика – такой вид кластера, в котором каждый документ подобен любому другому документу. Клика формируется тогда, когда возникает полный граф, т.е. полное соотношение подобия между всеми элементами.  А В    С Д  Исходными данными для метода является матрица подобия документа массива, которая заполняется коэффициентами подобия всех пар документов.  Матрица: S(Di , Dj) – диагональная квадратная и симметричная.  i = 1,N ; j = 1,N.  Пусть задано множество пар:  VDi = {(ti , wi)}  VDj = {(tj , wj)}  Коэффициент подобия документов определяется:  S(Di , Dj) = сумм(k =1,N)rk/N  r – отношение; N – мощность множества документов.  0, wi = 0 или wj = 0  rk = wi / wj в противномслучае  Чтобы задача решалась адекватно, вектора (\*) должны быть упорядочены по терминам, т.е. одни и те же термины должны быть записаны в одних и тех же позициях этих векторов. Исходная матрица, которая получена в результате расчётов, преобразуется в бинарную следующим образом: вводится некоторое пороговое значение T коэффициента подобия, и те коэффициенты, которые меньше его заменяются на 0, в противном случае на 1:  S(Di , Dj) < T , => 0  S(Di , Dj) > T , => 1  **Алгоритм:**  1.В класс или кластер включаются подгруппы порядка 2, т.е. те элементы, которые в отношении подобия установлены на паре.  2.Из подгруппы порядка 2 получают подгруппу порядка 3 по следующему правилу: если есть подгруппы (Di , Dj) , (Di , Dp) , (Dj , Dp), то получаем: (Di , Dj , Dp) и подгруппы из исходного списка исключаются.  3.Из подгруппы порядка p формируют подгруппу порядка (p+1),т.е. (Di , Dj , … , Dp) => (Di , Dj , … ,  ***Вопрос 33(продолжение).***  ***Последовательность.***  Это свойство гарантирует, что пользователь, освоивший работу в одной части системы не запутается, работая в другой её части.  *Выражается в 3-х явлениях:*   1. Последовательность в построении фраз. Т.е. вводимые коды или команды в системе всегда трактуются одинаково; 2. Последовательность в использовании форматов данных - аналогичные поля всегда представляются в одном формате (противоречит требованию гибкости); 3. Последовательность в размещении данных на экране.   *Рекомендуется следующий шаблон для оформления экрана:*   1. Вверху в 2-х, 3-х строках помещается заголовок и данные о состоянии системы; 2. Далее, под заголовком размещается область для вывода справочных сообщений; 3. Основная область – для рисования или для ввода данных; 4. Ниже – область для вывода сообщений об ошибках; 5. Описание функциональных клавиш.   ***Краткость.***  Требует от пользователя ввода минимума информации. Это, с одной стороны, убыстряет работу системы, а, с другой, приводит к появлению ошибок.  *Рекомендации:*   1. Не следует запрашивать информацию, которую следует сформировать автоматически; 2. Информация не должна выводится сразу же, только потому, что она стала доступна системе. Она должна выводится только в том объёме, который требуется пользователю и в нужном для него формате.   *Поддержка пользователя* – мера помощи, которую система оказывает пользователю при работе с ней.  *Эта поддержка выражается в 3-х видах:*   1. Инструкции пользователя. Выводятся в виде подсказок или справочной информации. При этом справочная информация должна быть контекстной, своевременной и доступной в любой точке диалога. Помимо внутрисистемной существует внешняя справочная информация, которая сопровождает текст в виде бумажного носителя. Там указывается 5 моментов:  * Общий обзор, в котором описывается назначение системы, основные понятия предметной области, необходимые для оценки системы, связанные с этими понятиями принципы работы системы; * Как начать работу с системой; * Сведения о поведении пользователя при выходе системы или отдельных частей из строя; * Пример работы с системой; * Ограничения на систему.  1. Сообщения об ошибках. Хорошее сообщение об ошибке должно отвечать следующим требованиям:  * Должно быть изложено в терминах, понятных пользователю; * Нужно точно определить причину ошибки; * Должно пояснять, как исправить ошибку; * Должно быть своевременным, пока не проделаны вещи, которые необратимы.  1. Подтверждения каких-либо действий системы.   ***Гибкость*** - мера того, насколько хорошо диалог соответствует различным уровням подготовки и производительности труда пользователя. Гибкость называют свойством адаптивности системы.  Существует 3 системы, которые характеризуют её гибкость:  ***37.Типы диалогов.***  *4 типа диалога:*   1. **вопрос – ответ;** 2. **меню;** 3. **командный язык;** 4. **экранные формы.**   ***Вопрос – ответ.*** Самая старая форма ведения диалога. Используется в экспертных системах, в информационно – поисковых системах к фактографическим или документальным базам данных.  3 вида диалога в режиме “вопрос – ответ”:   1. Диалог с ограничениями на предметную область. Форма запроса – произвольна (ограничений нет), а лексика запроса строится на базе 2-х словарей. 1-й содержит функциональные слова, которые либо означают характер задачи, которую нужно решить, либо носят вспомогательный характер, т.е. те запросы с которыми пользователь обращается к БД. Эти функциональные слова являются ключевыми, смысл их жёстко регламентирован. 2. 2-й словарь содержит специфические термины, которые характеризуют данную предметную область и, как правило, являются именами полей с записями базы данных. 1-е ограничение: если существуют надёжные окончания, то каждое слово из запроса нужно спроецировать на слова из словаря (где максимальное пересечение, то и брать). 2-е ограничение в рамках диалога – ограничение на язык.   Требования:   * Запрос или задание формируется с помощью фраз естественного языка, каждая из которых описывает элемент, операцию, которую надо выполнить. * Каждое предложение должно начинаться с функционального слова, определяющего нужное действие. * При формулировке условий поиска каждое значение поля БД должно предваряться названием этого поля.  1. 3-я форма – естественно языковая без ограничений.  * Этот диалог применяется тогда, когда диапазон либо слишком велик, либо вообще не определён. * Последующий запрос зависит от предыдущего, т.е. этот диалог нельзя заранее описать некоторым сценарием.   ***Меню*** – ориентированный диалог.  Здесь у пользователя есть список вариантов ответа и он выбирает нужный номер.  Виды меню:  1.  2. С использованием мнемонических обозначений опций (Norton Commander);  3. Блоковое;  4. Строчное меню;  5. Меню в виде пиктограмм.  *Требования к меню:*   1. Каждое меню должно содержать 5-6 опций; 2. При большом числе различных вариантов их надо группировать (подменю); 3. Пункты меню должны следовать в естественном порядке или по алфавиту. 4. *Применение меню:* 5. Диапазон возможных ответов невелик и они все известны заранее и могут быть представлены явно; 6. Когда пользователю необходимо видеть сразу все опции для выбора оптимальной, чтобы оценить все возможные варианты; 7. Когда пользователь неопытен.   ***40.Метод нисходящего синтаксического анализа(СА).***  Нисходящий СА (развёртка) – дерево разбора строится от корней к листьям.  СА методом развёртки. Здесь делается предположение, что исходное предложение уже принадлежит языку, а следовательно к ней применяется 1-я продукция грамматики, в которой левая часть является начальным символом грамматики. Этот шаг является 1-м шагом алгоритма развёртки. Введём здесь понятие элемента развёртки, роль которого на 1-м шаге правая часть продукции.  2-й шаг: из элемента развёртки выбирается крайний слева нетерминальный символ. Нетерминальный символ заменяется правой частью продукции с соответствующей левой частью того же списка продукции. Управление передаётся началу 2-го шага этого алгоритма. Если цепочка не содержит нетерминальных символов, она сравнивается с исходной анализируемой цепочкой. Если они совпадают, то конец алгоритма, иначе переход к шагу 3.  Шаг 3: разбор выполняется заново и при альтернативных вариантах продукции выбираются те, которые ранее не использовались. Т.е. выполняется разбор предложения фактически, по несколько другой схеме.  Если в грамматических правилах преобладают правила с одинаковыми левыми частями, оптимальнее выбирать восходящий разбор предложения и наоборот, если превалируют правила с альтернативными правыми частями нужно выбирать нисходящий разбор.  ***41.Метод восходящего синтаксического анализа(СА)***  Восходящий синтаксический анализ СА (свёртка) – дерево разбора строится от листьев к корню.  Алгоритм восходящего левостороннего СА:   1. Слева во фразе выделяется слово с морфологическими признаками (элемент свёртки). Если фраза пуста, по выполняется шаг 4, иначе следующий шаг; 2. Элемент свёртки сравнивается с правыми частями продукций. Если его совпадение, то элемент свёртки заменяется на левую часть продукции и выполняется передача на начало шага 2. Если совпадений нет выполняется следующий шаг. 3. Выбирается элемент из стека. Если при этом стек пуст, то элемент свёртки помещается в стек и выполняется шаг 1, иначе элемент из стека и элемент свёртки заменяются на понятия элемент свёртки и выполняется шаг 2. 4. Выполняется когда фраза обработана полностью. Здесь выбирается элемент из стека. Это должен быть начальный символ грамматики. В этом случае, если, к тому же, стек пуст, делаем 2 вывода: 5. Предложение принадлежит языку. 6. Построили его структуру, которая используется на этапе СеА.   Если условие не выполняется (стек не пуст), то либо предложение построено синтаксически неверно, либо при свёртке были использованы не те продукции. В этом случае возвращаются на тот шаг СА, где была возможность выбора альтернативной продукции.  ***45.Задача управления доступом.***  запрос  Ввод пароля  При решении этой задачи выделяют 3 понятия:   1. Субъект – пользователь, который выполняет некоторые операции над данными; 2. Объект операции доступа – те данные, к которым выполняется доступ; 3. Вид операции доступа.   В общем случае различают 3 вида доступа:   1. Неограниченный доступ ко всем данным; 2. Неограниченный доступ к группе данных; 3. Ограниченный доступ к группе данных. С ограничением как у субъекта, так и по видам операций.   **Подход к управлению доступом.**   1. Использование ключей и замков (идентификация и аутентификация пользователя). Идентификация – характеристики пользователя, которые его определяют. Идентификация позволяет определить себя пользователю, сообщив своё имя. Замок (средство аутентификации) позволяет 2-йстороне (ПК) убедиться что субъект – тот, за кого себя выдаёт.   Способы идентификации:  Пароль, личный идентификационный номер или криптографический ключ и т.д.;  Личная карточка;  Голос или отпечатки пальцев;  Нечто, ассоциируемое с субъектом.  Наиболее распространённый способ – использование паролей, когда аутентификация реализуется в программном обеспечении. Символически алгоритм аутентификации можно представить:  Выход в зависимости от  количества пользователей  Преимущество этого метода – гибкость, доступность и простота реализации.  Недостатки:  Нужно многократно вводить пароль;  Изменение замка оказывает влияние на большое число пользователей;  Замок управления может реализовать только то лицо, которое разрабатывает данное ПО.  Повышение надёжности парольной защиты осуществляется:  Пароль должен быть не слишком коротким и использовать различные символы;  Периодически менять пароли;  Если пароли не встроены в программу, а реализованы в виде файла, то ограничить доступ к файлу паролями;  Если возможны ошибки в пароле, то нужно ограничить число повторных вводов пароля;  Использование генераторов паролей, которые позволяют формировать благозвучные (хорошо запоминающиеся) пароли.   1. Если в качестве идентификатора используется личная карточка, для аутентификации используется устройство, называемое токеном – устройство, владение которым позволяет определить подлинность пользователя.   2 вида токенов:  Пассивные (токены с памятью);  Активные (интеллектуальные) токены.  Наиболее распространенными в силу дешевизны являются токены 1-го класса. Это устройства с клавиатурой и процессором, а личная карточка снабжена магнитной полосой. При использовании этого токена пользователь с клавиатуры вводит свой идентификационный номер, который проверяется процессором на совпадение с карточкой и далее проверяется подлинность самой карточки. Недостатки: дороговизна, использование специальных устройств чтения, неудобство использования. | ***22.Кластерные текстовые файлы.***  Документы разбиваются на родственные группы, которые называют кластерами или классами. Каждый класс описывается центроидом (профилем) и вектор запроса прежде всего сравнивается с центроидами класса.  Преимущества:   * Возможен быстрый поиск, т.к. число классов, как правило, невелико; * Возможно интерактивное сужение (расширение) поиска за счёт исключения или добавления дополнительных кластеров.   Недостатки:   * Необходимость формировать кластеры; * Необходимость введения файла центроидов; * Дополнительный расход памяти для файла центроидов или профилей.   ***23.Основные способы определения центроидов.***  1. Логический профиль (обозначается Р1).  Заданы вектора документов в следующем виде:  VDi = (d1 , d2 , d3 , … , dt), где t – число индексационных терминов, выделенных во всех документах нашего массива. Тогда логический профиль определяется как результат с логической “или”:  Р1 = VD1 v VD2 v VD3 v …  1, если термин tk входит в вектор документа Di;  dk (k = 1,t) = 0, в противном случае.  2.Профиль частотного документа (Р2).  Исходные данные также вектора документов. Исходная информация задана прошлой матрицей, а в формуле используются не логическое сложение, а арифметическое:  3.Профиль частотности термина (Р3).  Здесь используются веса терминов в документах:  VDi = (w1 , w2 , w3 , … , wt), где wi – веса терминов входящих в вектор документа VDi.  В вычислении Р3 участвуют веса.  Р3 (k = 1,t) = сумм(j = 1,N) wik , где j – порядковый номер документа; N – число документов кластера; t – число индексационных терминов в массиве; k – порядковый номер термина.  ***24. Логический способ определения центроидов.***   1. Логический профиль (обозначается Р1).   Заданы вектора документов в следующем виде:  VDi = (d1 , d2 , d3 , … , dt), где t – число индексационных терминов, выделенных во всех документах нашего массива. Тогда логический профиль определяется как результат с логической “или”:  Р1 = VD1 v VD2 v VD3 v …  1, если термин tk входит в вектор документа Di;  dk (k = 1,t) = 0, в противном случае.  ***25.Профиль частотности документа.***  Исходные данные также вектора документов. Исходная информация задана матрицей, а в формуле используются не логическое сложение, а арифметическое:  ***26.Профиль частотности термина.***  Здесь используются веса терминов в документах:  VDi = (w1 , w2 , w3 , … , wt), где wi – веса терминов входящих в вектор документа VDi.  В вычислении Р3 участвуют веса.  Р3 (k = 1,t) = сумм(j = 1,N) wik , где j – порядковый номер документа; N – число документов кластера; t – число индексационных терминов в массиве; k – порядковый номер термина.  ***Вопрос 28(окончание).***  Dp+1), если существуют пары: (Di , Dp+1) , (Dj , Dp+1) , … , (Dp , Dp+1).   1. Алгоритм заканчивается, когда дальнейшее слияние невозможно.   Недостаток метода: образуется большое число кластеров.  **29.Метод одной связи.**  Здесь классы представляются документами, между которыми установлены отношения подобия, которые подчиняются следующему требованию: между двумя документами устанавливается связь при выполнении одного из следующих условий: существуют Di , Dj,   1. S(Di , Dk) , S(Dk , Dj); 2. S(Di , Dk) , S(Dk , Dm) , S(Dm , Dj); 3. Самое сильное требование: S(Di , Dk) – если в матрице подобия уже существует подобие.   Если одно из условий выполняется, то документы в одном классе.  ***30.Кластеризация вокруг выборочных документов.***  Вместо построения матрицы подобия используют плотность пространства некоторых документов. В качестве возможных центров кластера выступают те документы, которые по результатам расчётов оказались расположенными в плотных зонах пространства. Все документы в данном методе делятся на 3 класса:   * Документы, уже включённые в кластеры; * Документы, ещё не подвергшиеся исследованию(не включенные в кластеры); * Свободные документы. Те документы, относительно которых делалась попытка включения в кластер, но она закончилась неудачей.   Берём документ, пробуем включить его в кластер. Если не получается, то заносим его в множество свободных документов. Далее из свободного множества пытаемся подключить документы к кластеру.  *Алгоритм:*   1. Выбирается очередной, не включённый в кластеры документ и считается возможным центром кластера; 2. Рассчитываются коэффициенты подобия между этим документом, документами свободного типа и документами, не включёнными в классы; 3. Плотность считается достаточной, если: 4. существует по меньшей мере n1 документов, коэффициенты подобия которых по отношению к выбранному документу превышает некоторое пороговое значение T1; 5. Существует по крайней мере n2 документов, коэффициенты подобия (КП) которых, по отношению к исследуемому документу превышают пороговое значение T2; 6. Если между n2 и T2 существует некоторое соотношение: n2>=n1; T2<=T1, тогда считается, что выбранный документ - “центр массы”, вокруг которого собраны другие документы. Если плотность недостаточна (если одно из условий не выполняется), то документ относят к свободным и аналогичным образом исследуют следующий документ из множества не включенных в кластеры. Если плотность достаточна, то формируют кластер, включающий все свободные документы и не включённые ранее в кластеры документы, для которых КП с исследуемым документом не меньше порогового значения T3. Для нового кластера строится вектор центроида, и все документы, включённые в этот кластер и те, которые включены в остальные кластеры сопоставляются с данным центроидом. Это сопоставление служит для отнесения каждого документа к одной из следующих категорий:   Вопрос 33(окончание).   1. Фиксированная адаптация. При этом пользователь сам явно выбирает свой уровень подготовки; 2. Полная адаптивность. Здесь диалоговая система строит модель пользователя, которая меняется автоматически по ходу работы системы. 3. Косметическая. Занимает промежуточное место между фиксированной полной и достигается использованием специальных приёмов:  * Приём использования сокращений (md – mkdir , cd – chdir , \*.bak); * Синонимы – пользователь выбирает то, что ему ближе;   Использование объектов по умолчанию и макросы.  ***34.Естественность и последовательность как критерии хорошего диалога.***  ***Естественность*** – свойство диалога, в соответствии с которым пользователю не нужно менять свои традиционные способы решения задачи.   1. *Свойство включает следующие аспекты:* 2. Диалог должен вестись на родном языке пользователя; 3. Стиль ведения диалога должен быть разговорным, т.е. имеется в виду краткость; 4. Фразы не должны требовать дополнительных пояснений; 5. Допускается использование жаргона пользователя; 6. Порядок ведения диалога должен соответствовать порядку, которым обычно пользователь обрабатывает информацию; 7. Должна быть исключена предварительная обработка данных перед их вводом в систему.   ***Последовательность.***  Это свойство гарантирует, что пользователь, освоивший работу в одной части системы не запутается, работая в другой её части.  *Выражается в 3-х явлениях:*   1. Последовательность в построении фраз. Т.е. вводимые коды или команды в системе всегда трактуются одинаково; 2. Последовательность в использовании форматов данных - аналогичные поля всегда представляются в одном формате (противоречит требованию гибкости); 3. Последовательность в размещении данных на экране.   *Рекомендуется следующий шаблон для оформления экрана:*   1. Вверху в 2-х, 3-х строках помещается заголовок и данные о состоянии системы; 2. Далее, под заголовком размещается область для вывода справочных сообщений; 3. Основная область – для рисования или для ввода данных; 4. Ниже – область для вывода сообщений об ошибках; 5. Описание функциональных клавиш.   ***Вопрос 37(окончание).***  ***Командный язык*** (MS – DOS)***.***  Используется для организации диалога с операционной системой. Требует хорошей квалификации пользователя и команды должны нести смысловую нагрузку.  Параметры командного языка могут быть 2-х типов: позиционные и ключевые.  *Применение такого диалога:*   1. Число значений для ввода мало и их можно запомнить; 2. Задача не требует много данных на вводе; 3. Опытность пользователя.   ***Экранная форма.***  Позволяет получить сразу всю информацию от пользователя, поскольку он отвечает сразу на несколько вопросов. И, следовательно:  Она быстрее работает;  Может работать с более широким диапазоном данных, чем меню;  Может использоваться пользователем любой квалификации.  Широко используется в Windows.  ***38.Задача морфологического анализа естественно-языковых текстов.***  Морфологический анализ (МА) – выделяет гипотетические основы слов и приписывает им различные грамматические категории. Может включать в себя морфографический анализ;  МА выполняется 2-мя способами:   1. Декларативный: т.е. словарь системы содержит все всевозможные словоформы языка, с приписанными им грамматическими характеристиками. Словоформы – все возможные модификации, которые существуют в языке; 2. Процедурный способ проведения МА – когда МА выделяет основу слова по словоизмерительным аффиксам и приписывает этой основе необходимые грамматические характеристики.   Рассмотрим 2-й способ. Состоит из 2-х шагов:   1. Определение морфологического типа и части речи; 2. Получение списка гипотетических основ и знаний грамматических категорий.   Для реализации 1-го шага используется таблица словоизмерительных аффиксов, которая приведена в приложении (Л – любой)  ***39.Задача синтаксического анализа естественно-языковых текстов.***  Синтаксический анализ (СА). Преследует 2 цели:   1. Проверка правильности построения фраз, т.е. соответствие её правилам языка; 2. Построение синтаксической структуры фразы, которая используется при выполнении следующей фазы – семантического анализа (СеА);   СА использует правила сочетаемости грамматических категорий, например, по числу, падежу и правила построения типичных языковых конструкций.  В зависимости от направления СА он может быть левосторонним и правосторонним. В 1-ом случае исходная фраза анализируется слева направо, во втором – справа налево.  2-й подход классификации: в зависимости от правил применения продукций различают:   1. восходящий СА (свёртка) – дерево разбора строится от листьев к корню; 2. нисходящий СА (развёртка) – дерево разбора строится от корней к листьям.   ***42.Постановка задачи семантического анализа .***  Выявляет смысл предложения и отвергает те фразы, которые бессмысленны для данной предметной области. При выполнении СеА используется база знаний(БЗ) по предметной области, представленная как правила для этих задач в виде семантических сетей или фреймов.  Грамматика фразы: “Зелёные идеи яростно спят.”.  Это результат СА структуры.  Семантический анализатор на основе дерева строит следующую семантическую сеть:    Идеи Что делают Спят  Какие как    Зелёные Яростно  Эти связи возникли:   1. Из-за свёртки глагола и наречия в группу глагола (нагружается весом “как”); 2. Из-за свёртки прилагательного и существительного в группу существительного и нагружается весом “какие”, т.к. прилагательное в предложении как правило является определением; 3. Из-за свёртки группы существительного и группы глагола в предложение и нагружается весом “что делают”, т.к. эта связка связывает существительное и глагол в предложении.   Построенная семантическая сеть называется ситуативной и возникает в системе по мере прихода и анализа естественно-языковых фраз. Одновременно с ситуативными фразами, в системе существует БЗ, включающая парадигматические отношения между понятиями предметной области, которые тоже представлены некоторой семантической сетью.  Рисунок в приложении.  Между этими вершинами существуют связи, которые носят аналоговый характер. Наша ситуативная семантическая сеть проецируется на БЗ, которая присутствует в системе. Возникают следующие противоречия:  “идеи” и “спят” относятся к таким классам, где существует отношение с весом “никогда”. Аналогично между “идеи и зелёные” и “спят и яростно ” никогда не установятся семантические отношения. Т.о. в исходной фразе найдены 3 противоречия, в соответствии с которыми она отвергается как семантически некорректная.  При фреймовом подходе:   1. К решению задач СеА, с каждым слотом связывается правило, по которому формируется его значение; 2. Выбор конкретного фрейма при анализе определяется некоторыми ключевыми словами во фразе, роль которых как правило играют глаголы.   С СеА тесно связана интерпретация понятий. Если фраза корректна, то каждому понятию даём поле БД, например, у нас было бы 4 поля (А,В,С,Д), т.е. заполнение информации в БД.  ***Вопрос 45(окончание).***  Активные токены имеют собственную вычислительную мощность, т.е. способность не только к памяти, но и анализу. Пользователь вводит свой идентификационный номер, а дальнейшие действия токена определяются в зависимости от его вида:  Определяет статический обмен паролями. Пользователь вводит пароль, а затем он проверяется ПК;  Динамическая организация пароля. В ПК и в токене установлены синхронно работающие системы генерации паролей.  Запросно-ответная система. ПК выдаёт случайное число, которое преобразуется криптографическим механизмом, встроенным в токен. После чего этот пароль возвращается в ПК. Пользователь вводит его либо вручную, либо через электронный интерфейс.  Достоинство: обеспечивают ежедневно меняющийся пароль.  Недостаток: дороговизна.   1. Голос или отпечатки пальцев (сетчатка глаза). Средства биологической аутентификации очень сложны и используются в специальных случаях, когда объекту требуется дополнительное обеспечение безопасности.   Аутентификация путём определения координат. Целесообразно использовать для распределённых систем (клиент-сервер). Сервер аутентификации на основании положения спутника по имени ПК в сети определяет его географическое местоположение с точностью до метра.  **2-й подход управления доступом:**  Таблица управления доступом (таблица безопасности).  Поддерживается в локальных или распределённых БД и могут поддерживаться ОС.  **Структура таблицы показана в приложении.**  Определяются операции, которые доступны.  Достоинства: возможность построения таблицы пользователя, простота изменения, отсутствие необходимости многократного указания ключей.  **Протоколирование и аудит.**  3-й подход к управлению доступом.  Протоколирование – сбор и накопление информации о событиях, происходящих в информационной системе предприятия.  Аудит – периодический анализ накопленной информации.  Цели подхода:   * Обеспечение подотчётности пользователей и администрации; * Обнаружение попыток нарушения ИБ.   **Шифрование или криптографическое кодирование.**  **Экранирование**.  Скрытие данных на экране (физически информация остаётся, но ёе не видно). | ***27.Постановка задачи кластеризации документов.***  Задача кластеризации состоит в том, чтобы разнести документы по группам таким образом, чтобы документы одной группы были достаточно сходны друг с другом, так, чтобы индивидуальными различиями можно было пренебречь.  **1.Нахождение КЛИК.**  Клика – такой вид кластера, в котором каждый документ подобен любому другому документу. Клика формируется тогда, когда возникает полный граф, т.е. полное соотношение подобия между всеми элементами.  А В    С Д  Исходными данными для метода является матрица подобия документа массива, которая заполняется коэффициентами подобия всех пар документов.  Матрица: S(Di , Dj) – диагональная квадратная и симметричная.  i = 1,N ; j = 1,N.  Пусть задано множество пар:  VDi = {(ti , wi)}  VDj = {(tj , wj)}  Коэффициент подобия документов определяется:  S(Di , Dj) = сумм(k =1,N)rk/N  r – отношение; N – мощность множества документов.  0, wi = 0 или wj = 0  rk = wi / wj в противномслучае  Чтобы задача решалась адекватно, вектора (\*) должны быть упорядочены по терминам, т.е. одни и те же термины должны быть записаны в одних и тех же позициях этих векторов. Исходная матрица, которая получена в результате расчётов, преобразуется в бинарную следующим образом: вводится некоторое пороговое значение T коэффициента подобия, и те коэффициенты, которые меньше его заменяются на 0, в противном случае на 1:  S(Di , Dj) < T , => 0  S(Di , Dj) > T , => 1  **2.Метод одной связи.**  Здесь классы представляются документами, между которыми установлены отношения подобия, которые подчиняются следующему требованию: между двумя документами устанавливается связь при выполнении одного из следующих условий: существуют Di , Dj,   1. S(Di , Dk) , S(Dk , Dj); 2. S(Di , Dk) , S(Dk , Dm) , S(Dm , Dj); 3. Самое сильное требование: S(Di , Dk) – если в матрице подобия уже существует подобие.   Если одно из условий выполняется, то документы в одном классе.  **3.Кластеризация вокруг выборочных документов.**  Вместо построения матрицы подобия используют плотность пространства некоторых документов. В качестве возможных центров кластера выступают те документы, которые по результатам расчётов оказались расположенными в плотных зонах пространства. Все документы в данном методе делятся на 3 класса:   * Документы, уже включённые в кластеры; * Документы, ещё не подвергшиеся исследованию(не включенные в кластеры); * Свободные документы. Те документы, относительно которых делалась попытка включения в кластер, но она закончилась неудачей.   Берём документ, пробуем включить его в кластер. Если не получается, то заносим его в множество свободных документов. Далее из свободного множества пытаемся подключить документы к кластеру.  ***Вопрос 30(окончание).***  Документы, КП которых превышает пороговое значение T3, включаются в состав нового кластера;   * Документы, КП которых меньше или равно T3, но больше некоторого порогового значения T4, включаются в множество свободных документов; * Документы, КП которых не больше T4, называются не включенными в кластеры.   Этот процесс повторяется до тех пор, пока все документы не будут отнесены к разряду свободных или не будут включены в какой-либо кластер.   1. Рассчитывается КП всех свободных документов со всеми центроидами кластеров и там, где подобие окажется максимальным и относят каждый свободный документ.   ***31.Коррекция кластеров сверху вниз.***  В начале строятся один или несколько очень больших кластеров, которые затем разбиваются на более мелкие.  *Способы выбора исходных классов:*   * В качестве центров классов используются случайные документы; * Классом с именем i можно считать множество документов, в векторах которых находится термин i; * В качестве исходных классов принимаются все документы, признанные релевантными некоторому запросу по результатам предыдущих поисковых операций.   *Процесс коррекции кластеров:*   * Вычисляется КП между каждым документом и каждым центроидом кластера; * Кластеры переопределяются путём отнесения документов к тем из них, по отношению к которым, они имеют наибольшее подобие; * Формируются центроиды новых кластеров.   Эти 3 шага выполняются до тех пор, пока:   * Будет необходимость в изменениях; * Чтобы процесс не был бесконечным, он выполняется в заданное число итераций.   ***32.Однократная кластеризация.***  Документы рассматриваются в произвольном порядке и каждый документ либо относится к существующему классу, если КП достаточен, либо образует новый кластер.  “+”: каждый документ обрабатывается только 1 раз, => требует мало времени.  “-”: состав и структура классов существенно зависит от порядка рассмотрения документов.  ***33.Основные критерии хорошего диалога.***   1. **Естественность;** 2. **Последовательность;** 3. **Краткость;** 4. **Поддержка пользователя;** 5. **Гибкость.**   ***Естественность*** – свойство диалога, в соответствии с которым пользователю не нужно менять свои традиционные способы решения задачи.   1. *Свойство включает следующие аспекты:* 2. Диалог должен вестись на родном языке пользователя; 3. Стиль ведения диалога должен быть разговорным, т.е. имеется в виду краткость; 4. Фразы не должны требовать дополнительных пояснений; 5. Допускается использование жаргона пользователя; 6. Порядок ведения диалога должен соответствовать порядку, которым обычно пользователь обрабатывает информацию; 7. Должна быть исключена предварительная обработка данных перед их вводом в систему.   ***35.Краткость и поддержка пользователя как критерии хорошего диалога.***  ***Краткость.***  Требует от пользователя ввода минимума информации. Это, с одной стороны, убыстряет работу системы, а, с другой, приводит к появлению ошибок.  *Рекомендации:*   1. Не следует запрашивать информацию, которую следует сформировать автоматически; 2. Информация не должна выводится сразу же, только потому, что она стала доступна системе. Она должна выводится только в том объёме, который требуется пользователю и в нужном для него формате.   *Поддержка пользователя* – мера помощи, которую система оказывает пользователю при работе с ней.  *Эта поддержка выражается в 3-х видах:*   1. Инструкции пользователя. Выводятся в виде подсказок или справочной информации. При этом справочная информация должна быть контекстной, своевременной и доступной в любой точке диалога. Помимо внутрисистемной существует внешняя справочная информация, которая сопровождает текст в виде бумажного носителя. Там указывается 5 моментов:  * Общий обзор, в котором описывается назначение системы, основные понятия предметной области, необходимые для оценки системы, связанные с этими понятиями принципы работы системы; * Как начать работу с системой; * Сведения о поведении пользователя при выходе системы или отдельных частей из строя; * Пример работы с системой; * Ограничения на систему.  1. Сообщения об ошибках. Хорошее сообщение об ошибке должно отвечать следующим требованиям:  * Должно быть изложено в терминах, понятных пользователю; * Нужно точно определить причину ошибки; * Должно пояснять, как исправить ошибку; * Должно быть своевременным, пока не проделаны вещи, которые необратимы.  1. Подтверждения каких-либо действий системы.   ***36.Гибкость как критерий хорошего диалога.***  ***Гибкость*** - мера того, насколько хорошо диалог соответствует различным уровням подготовки и производительности труда пользователя. Гибкость называют свойством адаптивности системы.  Существует 3 системы, которые характеризуют её гибкость:   1. Фиксированная адаптация. При этом пользователь сам явно выбирает свой уровень подготовки; 2. Полная адаптивность. Здесь диалоговая система строит модель пользователя, которая меняется автоматически по ходу работы системы. 3. Косметическая. Занимает промежуточное место между фиксированной полной и достигается использованием специальных приёмов:  * Приём использования сокращений (md – mkdir , cd – chdir , \*.bak); * Синонимы – пользователь выбирает то, что ему ближе;   Использование объектов по умолчанию и макросы.  Вопрос 39(окончание).  Алгоритм восходящего левостороннего СА:   1. Слева во фразе выделяется слово с морфологическими признаками (элемент свёртки). Если фраза пуста, по выполняется шаг 4, иначе следующий шаг; 2. Элемент свёртки сравнивается с правыми частями продукций. Если его совпадение, то элемент свёртки заменяется на левую часть продукции и выполняется передача на начало шага 2. Если совпадений нет выполняется следующий шаг. 3. Выбирается элемент из стека. Если при этом стек пуст, то элемент свёртки помещается в стек и выполняется шаг 1, иначе элемент из стека и элемент свёртки заменяются на понятия элемент свёртки и выполняется шаг 2. 4. Выполняется когда фраза обработана полностью. Здесь выбирается элемент из стека. Это должен быть начальный символ грамматики. В этом случае, если, к тому же, стек пуст, делаем 2 вывода: 5. Предложение принадлежит языку. 6. Построили его структуру, которая используется на этапе СеА.   Если условие не выполняется (стек не пуст), то либо предложение построено синтаксически неверно, либо при свёртке были использованы не те продукции. В этом случае возвращаются на тот шаг СА, где была возможность выбора альтернативной продукции.  СА методом развёртки. Здесь делается предположение, что исходное предложение уже принадлежит языку, а следовательно к ней применяется 1-я продукция грамматики, в которой левая часть является начальным символом грамматики. Этот шаг является 1-м шагом алгоритма развёртки. Введём здесь понятие элемента развёртки, роль которого на 1-м шаге правая часть продукции.  2-й шаг: из элемента развёртки выбирается крайний слева нетерминальный символ. Нетерминальный символ заменяется правой частью продукции с соответствующей левой частью того же списка продукции. Управление передаётся началу 2-го шага этого алгоритма. Если цепочка не содержит нетерминальных символов, она сравнивается с исходной анализируемой цепочкой. Если они совпадают, то конец алгоритма, иначе переход к шагу 3.  Шаг 3: разбор выполняется заново и при альтернативных вариантах продукции выбираются те, которые ранее не использовались. Т.е. выполняется разбор предложения фактически, по несколько другой схеме.  Если в грамматических правилах преобладают правила с одинаковыми левыми частями, оптимальнее выбирать восходящий разбор предложения и наоборот, если превалируют правила с альтернативными правыми частями нужно выбирать нисходящий разбор.  ***43.Основные аспекты информационной безопасности.***  Под безопасностью БД понимается их защита от случайного или преднамеренного разрушения, искажения или утечки. Решение этой проблемы относится к организационному виду обеспечения информационных систем.  **Аспекты ИБ.**   1. Идеологический аспект обеспечения ИБ. Состоит в разъяснении, внушении работникам фирмы правил в необходимости обеспечения ИБ фирмы. 2. Управленческий аспект – разработка различных указаний, распоряжений, регламентирующих права и обязанности работников по обеспечению ИБ фирмы. 3. Организационный. Связан с созданием специального служебного обеспечения ИБ и принятия соответствующих защитных мер. 4. Программно-технический аспект. Включает 2 направления: 5. Управление доступом. Обеспечивает защиту от несанкционированного доступа; 6. Управление целостностью. Обеспечивает защиту от неверных изменений и разрушений. Это управление рассматривается в 3-х аспектах:  * Обеспечение достоверности – предупреждение возможных ошибок в значениях данных из-за ошибок в управлении; * Обеспечение параллелизма в выполнении операций над информацией. Состоит в том, что целостность информации не нарушается при одновременном выполнении нескольких операций; * Восстановление данных. При программных или аппаратных сбоях необходимо обеспечить быстрое восстановление данных и продолжение работы системы.   ***44.Постановка задачи программно-технического обеспечения информационной безопасности.***  Программно-технический аспект. Включает 2 направления:   1. Управление доступом. Обеспечивает защиту от несанкционированного доступа; 2. Управление целостностью. Обеспечивает защиту от неверных изменений и разрушений. Это управление рассматривается в 3-х аспектах:  * Обеспечение достоверности – предупреждение возможных ошибок в значениях данных из-за ошибок в управлении; * Обеспечение параллелизма в выполнении операций над информацией. Состоит в том, что целостность информации не нарушается при одновременном выполнении нескольких операций; * Восстановление данных. При программных или аппаратных сбоях необходимо обеспечить быстрое восстановление данных и продолжение работы системы.   ***46.Задача управления целостностью.***  Аспекты решения этой задачи:   1. Обеспечение достоверности. Как правило эта задача решается для структурированных БД и сводится к обеспечению значений ключевых и неключевых полей, во-первых, в соответствии с семантической предметной областью, во-вторых, в соответствии со смыслом понятия “ключевое поле” или “ключ”.   Характеристики данных, которые при этом вводятся носят структурный или семантический характер.   1. Структурные характеристики не имеют отношения к семантике предметной области и уникальны. 2 вида структурных характеристик:  * Значение ключевых полей должно быть непустым; * Адресные ссылки должны относиться к непустым записям.  1. Семантических характеристик можно ввести сколько угодно, в соответствии с предметной областью, т.е. выделяются с предметной областью. Чем больше этих характеристик, тем корректней данные, но тем дольше идёт обработка и заполнение БД. 2. Управление параллелизмом. Наиболее актуальна в распределённых системах. Эта задача возникает, когда к одним данным обращается несколько пользователей. Для того чтобы при выполнении параллельных задач не возникла задача некорректности получаемой информации вводится блокировка данных, если они запрашиваются процедурой, связанной с их модификацией.   Суть: если к данным х1 обращается транзакция (процедура) t1 с целью их модификации, она запрещает доступ к этим данным всем остальным транзакциям до тех пор, пока сама полностью не отработает. После этого выполняется разблокировка данных.   1. Восстановление данных. Задача возникает при аппаратных и программных сбоях. Решение задачи состоит в ведении системного журнала, в котором фиксируются все изменения, осуществляемые с БД. При возникновении сбоя, система откатывается в ближайшую точку, начиная с которой её повторяется (воспроизводится заново) до сбоя и далее. |

***2.Эволюция информационной технологии.***

Этапы эволюции современных ИТ:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Этап | Годы | Тип ЭВМ пользователь | Режим исп-ния ЭВМ | Осн. ср-ва связи с ЭВМ | Осн. ср-ва наложения инф. | Критерии эффективности | Ключ. Решения в обраб. инф. | Схемы информатизации |
| 1 | 50 г. | Большие ламповые ЭВМ Пользователь был программистом | Монопольный | Пульт упр-ния. | Перфокарты, -ле нты , магн. 2ленты | Машин. Ресурсы (быстродействие, оптим. время выполнения). | Англ язык прог- ния | ВЦ |
| 2 | 60 г. | Пакетный | Пульт, алф.-цифр. Дисплей | -- // -- магн. Диски | ОС, Оптимизированные трансляторы | ВЦ коллективного пользования, АСПУ, АСУТП |
| 3 | 70 г. | Мини ЭВМ, программист | Мон-ный, с разде  лением времени диалог. | Алф.-цифр. Дисплей | Человеч. Ресурсы.Трудоемкость сопров. И отл. Пр-мы | Интерактивные ОС, структурированные ЯП, компьютерные сети | АСНИ науч исс- ний, ГАП –гибкие авт – ные пр-ва, САПР, АРМ –раб места |
| 4 | 80 г. | Персональный, конечный пользователь. | Мон-ный, диалог. | Цветной граф. режим. | Оптич. Гибкие, жесткие | Трудность форм-ции проф-ных знаний | Технология автоматизации проф знаний | Интеллектуальные системы ППП (текстовые, граф.) |
| 5 | 90 г. | ПК в сети, конечный пользователь. | Коллект. Диалог. | Вирт. терминал обес-щий доступ к любому уровню в сети | Полнота и скорость доступа | Коллективный доступ к информационным ресурсам инф-ная безопасность | Internet |

Приложения:

Таблица для вопроса 38(***Задача морфологического анализа естественно-языковых текстов).***

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Аффикс | Часть  Речи | Морф. Типы (МТ) | Род | Число | Падеж | Примеры |
| Ы | Сущ. | 1, 2, 7 | М | М | И | Заводы |
| Ы | Сущ. | 13, 18 | Ж | М | И | Карты |
| Ы | Сущ. | 13, 18 | Ж | Е | Р | Улицы |
| Ы | Прил. | 9 | Л | М | И, В | Отцовы |
| Й | Сущ. | 8, 9 | М | Е | И | Герой |
| Й | Сущ. | 19, 20 | Ж | М | Р | Статуй |
| Й | Сущ. | 27 | С | М | Р | Зданий |
| ЫХ | Прил. | 1, 5, 6, 9 | Л | М | В, Р | Новых |
| Й | Прил. | 2, 3, 4, 8 | М | Е | И, В | Синий |

Рисунок к вопросу 42(Постановка задачи семантического анализа).

Иметь цвет

Никогда

находится в

Неодушевлённые состоянии пассивное иметь обстоятельства

Нематериальные никогда действие никогда активного образа

Предметы действия

идеи зелёные спят яростно

Таблица к вопросу 45(Задача управления доступом).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Объект1 | Объект2 | … | Объектn |
| Пользователь1 | Чтение(запись) |  |  | чтение |
| Пользователь2 |  |  |  |  |
| … |  |  |  |  |
| Пользовательn |  |  |  |  |