119270, Москва, Лужнецкая наб., д. 6,
стр.1, офис 214, ООО «ЭР СИ О»
Тел./факс: (495) 287-98-87
E-mail: info@rco.ru
http://www.rco.ru

Руководство разработчика
RCO News Clustering Engine

Москва, 2020

В содержание данного документа могут быть внесены изменения без предварительного уведомления. Названия организаций, имена и даты, используемые в качестве примеров, являются вымышленными, если не оговорено обратное.

Содержание

Введение

Программное обеспечение (ПО) **RCO News Clustering Engine** (далее Агрегатор новостной ленты) предназначено для связывания сообщений, описывающих одни и те же события, в кластеры – сюжеты.

Агрегатор новостной ленты использует алгоритмы разбора текста, взаимного взвешивания документов, кластеризации документов. При построении кластеров Агрегатор на каждой итерации рассматривает временной интервал, называемый окном кластеризации. Итерации повторяются со сдвигом окна кластеризации на заданный временной отрезок, называемый шагом кластеризации.

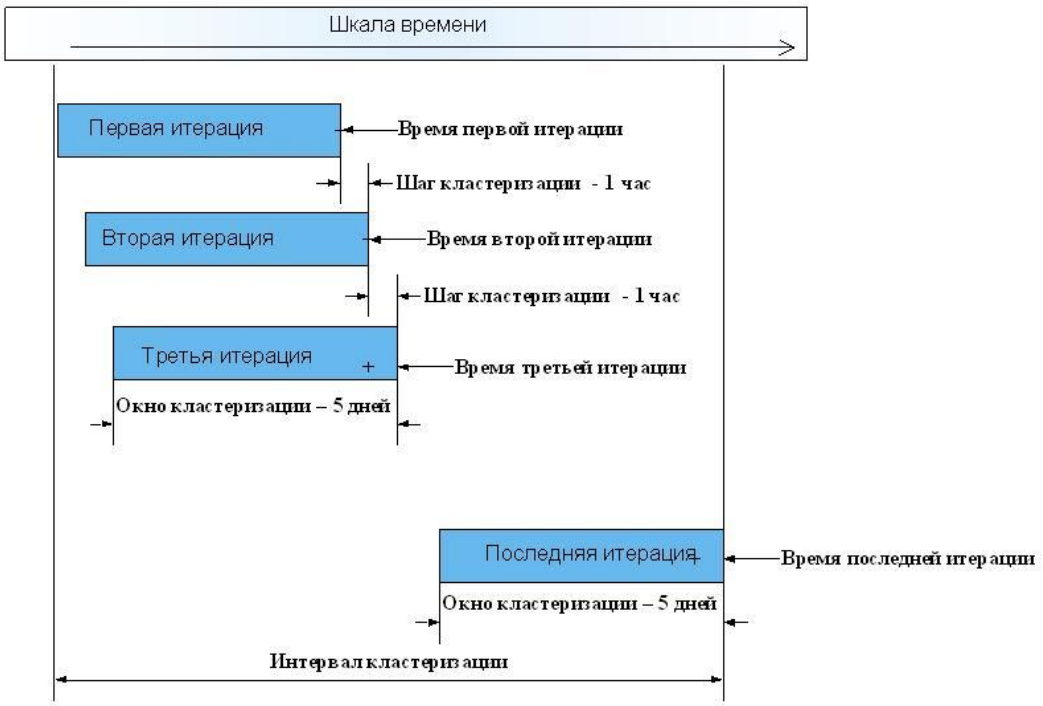


Рисунок 1 – Шаг, окно кластеризации

Агрегатор новостной ленты в штатном режиме предназначен для работы с СУБД (дополнительно предусмотрен режим отладки, в котором обращения к СУБД заменяются на работу с файловой системой). Входной информацией агрегатора являются документы новостной ленты, хранящиеся в базе данных новостей заданного формата. Агрегатор новостной ленты сохраняет результат кластеризации документов в базе данных новостей в специально разработанных таблицах. Результатом кластеризации является набор кластеров. Каждый кластер имеет набор документов, собственно образующих кластер, и набор терминов, характеризующих кластер. Указанные документы и термины имеют свой вес в кластере.

Помимо штатного режима, в котором происходит обработка новых документов, Агрегатор новостной ленты имеет также ретроспективный режим, в котором за явно указанный интервал проводится кластеризация документов с заданными окном и шагом кластеризации. В базе данных сохраняются результаты всех итераций кластеризации.

Внутренняя структура агрегатора допускает достаточно легкую замену одного или нескольких алгоритмов агрегатора на этапе компиляции и сборки ПО.

Конфигурация и состав программы

Программа представляет собой исполняемый файл (консольное приложение) целевой платформы, с которым статически связаны реализации алгоритмов кластеризации и библиотека анализа русского текста.

Файл	**Описание**
RCO_NCE_DevGuide.doc	Данный документ.
RCOAggregator	Исполняемый файл агрегатора.
RCOAggregator.log	Файл протокола.
RCOAggregator.ini	Файл настроек агрегатора, считываемый в момент старта ПО. Содержит следующие параметры: 1. Максимальное количество терминов, характеризующих кластер (параметр передается процедуре **`pkgrco_aggr_cluster_describing_terms`**): *TermRestrictionValue=10* – исходное значение 2. Пороговое значение близости документов. Пары документов, близость между которыми меньше указанного значения, не рассматриваются в процессе кластеризации. *IgnoreClusteringValue=0.001* – исходное значение Целая и дробная части значений параметров разделяются точками. 3. Момент времени, при наступлении которого агрегатор останавливается (проверка производится в момент окончания расчета итерации): *Until=DD/MM/YYYY-HH:MM* 4. Сохранять в БД или нет вычисленные расстояния между документами: *WriteSimilarities=0* 5. Продолжать последнюю успешную итерацию (если она есть) или работать независимо от результатов предыдущих запусков агрегатора (сохранит идентификаторы схожих кластеров для преемственности сюжетной линии): *ContinueLastIteration=1*.
RCOFXRu.dll	Библиотека для автоматического анализа русского текста, выделения терминов.
fx.ini	Настройки библиотеки анализа.
cb.ini, te.ini	Настройки алгоритма кластеризации.
dic*.*, ld*.*, obj*.*	Файлы с лингвистическим обеспечением для анализа текста.
RCOAggregator_schema.sql	Скрипт первоначального создания объектов БД агрегатора.

Библиотека автоматического анализа русского текста требует наличия русской кодовой страницы 1251.

Установка программы

1. Создать объекты схемы БД путём выполнения скриптов из каталога sql дистрибутива[1];

2. Привести **view RCO_AGGR_NEWS**, **RCO_AGGR_NEWS_RBR**, **RCO_AGGR_RUBRICATOR** к вашей собственной модели данных (документы, рубрики, связка документ рубрики);

3. Убедиться, что на машине, где будет развернут модуль агрегации новостей, есть правильно настроенный клиент БД;

4. Скопировать полностью папку **\Aggregator** на жесткий диск;

5. В файле **\Aggregator\AgregatorDB.cmd** задать в командной строке параметры присоединения к БД, например: `RCOAggregator.exe connect=пользователь/пароль@БД days=7 every=1 interval=1/1/2017-31/12/2020`;

6. Зарегистрировать агрегатор новостей как сервис операционной системы, отредактировав и выполнив файл **service_DB_install.cmd**;

7. Запустить сервис **RCONewsAggr**.

Запуск программы без использования БД

Доступен режим работы программы без подключения к базе данных, в котором входные данные считываются из файловой системы, туда же печатаются результаты работы, а промежуточные данные хранятся в оперативной памяти компьютера. Для запуска программы в этом режиме необходимо:

1. Разместить входные данные в каталоге, путь к которому задаётся параметром gz-dir в AgregatorHDD.cmd: данные входного потока новостей должны быть записаны по дням (один день – один файл) с именами вида docs_yyyymmdd, каждый файл заархивирован архиватором gzip в расширение gz, где yyyy – год, mm – месяц, dd – день числами. Структура файла – таблица с символом табуляции в качестве разделителя колонок и названий колонок в первой строке:

- DOCID – идентификатор новостного сообщения
- ISSUEDATE – дата публикации
- RECORDDATE – дата скачивания
- SOURCEID – идентификатор источника, опубликовавшего сообщение
- SOURCENAME – название источника
- TITLE – заголовок сообщение
- TEXT – текст сообщения

2. Задать в AgregatorHDD.cmd параметры запуска (описаны в разделе Запуск программы) и запустить его. Можно также установить программу в качестве сервиса, запустив service_HDD_install.cmd, и обращаться к этому сервису.

3. Результаты работы программы записываются в файл clusters_added_N.txt, где N – номер итерации, в виде таблицы с символом табуляции в качестве разделителя колонок: cluster_id – номер кластера, doc_id – идентификатор документа, входящего в кластер, weight – вес документа в кластере.

[1] Скрипты могут требовать адаптации в зависимости от используемой базы данных

Работа программы

Запуск программы

Запуск Агрегатора производится со следующими параметрами:

- `interval=`*dd/mm/yyyy-dd/mm/yyyy* – интервал кластеризации в днях и часах. Данный параметр определяет, в каком (штатном или ретроспективном) режиме запущен агрегатор. Если данный параметр задан (ретроспективный режим), агрегатор производит необходимое число итераций кластеризации по документам, хранящимся в базе и принадлежащих заданному интервалу. Если параметр не задан, агрегатор обрабатывает свежие новости (штатный режим);
- `days=`*число_дней* – ширина окна кластеризации в днях. Изначальное значение – 5 дней. Для тестовых запусков был выбран период 7 дней;
- `every=`*число_часов* – шаг кластеризации в часах. В штатном режиме данный параметр определяет, как часто агрегатор будет обрабатывать новые поступающие в БД документы. В ретроспективном режиме – на сколько сдвигать окно кластеризации при следующей итерации. Исходно – 1 час;
- `connect=`*строка_соединения_с_БД* – строка соединения с БД новостей в формате `имя_пользователя/пароль@строка_подключения_TNS`. Обязательный параметр. Необходимый набор полномочий для пользователя описан в разделе «Взаимодействие с базой новостей». Одновременно можно иметь только один работающий агрегатор в одной базе данных;
- `noalign` – в отсутствие иных установок, время итерации кластеризации выравнивается по границе часа, при указании данного параметра в штатном режиме выравнивания не происходит. В ретроспективном – данный параметр игнорируется и всегда производится выравнивание по границе часа;
- `once` – произвести только одну итерацию в штатном режиме. В ретроспективном режиме данный параметр игнорируется;
- `extract_only` – производятся только лингвистический анализ документов в ретроспективном режиме. В штатном режиме параметр игнорируется.
- `gz-dir` – путь к директории с новостными сообщениями в режиме работы без базы данных. Формат описан в разделе Запуск программы без использования БД.

Завершение программы

Корректное завершение программы производится посылкой сигнала `SIGINT` (Ctrl+C).

В ретроспективном режиме (`interval`) и режиме штатного однократного выполнения (*once*) программа остановится после выполнения кластеризации.

Протоколирование работы программы

Программа фиксирует время начала и окончания расчета каждой итерации кластеризации в таблице **`RCO_AGGR_RUNS`**. В случае отсутствия ошибок полю `STATUS` присваивается значение *'SUCCESS'*;

- На целевой платформе запуск/завершение/критические ошибки программы будем сохранять в системный журнал (*syslog* ());
- Программа фиксирует всю вышеперечисленную информацию, а также полные сообщения об ошибках и прочие трассировочные сообщения в файле **RCOAggregator.log**.

Краткое описание алгоритмов агрегатора

Система агрегации новостного потока состоит из 3 модулей:

1. Лингвистического анализа документов и выделения ключевых терминов;
2. Вычисления расстояния между документами в пространстве выделенных терминов;
3. Группировки документов, посвященных одним событиям, в кластеры и ведение сюжетной линии.

Работа всех модулей координируется в одном приложении. Промежуточные результаты (результаты разбора текстов, матрица близости документов, кластера) хранятся в базе.

Особенности функционирования модулей:

1. Лингвистический анализ документов и выделение ключевых терминов:

   – Производятся полный морфологический и синтаксический анализ документа;

   – Из текста выделяются не только слова, но и словосочетания, которые объединяются со словами через оператор «*ИЛИ*». Например, *ДОХОД|ДОХОД "VIEWSONIC"|МИРОВОЙ ДОХОД|МИРОВОЙ ДОХОД "VIEWSONIC"*;

   – Все служебные части речи отбрасываются. Малоинформативные, общеупотребительные слова фильтруются тезаурусом;

   – Словам приписываются семантические типы (имя персоны, название организации, продукта, география и т.д.);

   – В зависимости от семантического типа терминам приписывается вес;

   – Дополнительно идентифицируются специальные конструкции, такие как «*По информации агентства Reuters, ...*», а также придаточные части предложения. Такие конструкции не отражают сути новости, поэтому исключаются.

2. Вычисление расстояния между документами в пространстве выделенных терминов:

   – Помимо веса, приписываемого терминам на стадии разбора текстов, словам приписывается вес по TFiDF (TF – term frequency, IDF – inverse document frequency);

   – Термины сравниваются комплексно, с учетом словосочетаний. Так как словосочетания встречаются реже одинарных слов, iDF для них будет выше, поэтому

пересечение текстов по словосочетаниям дает более сильное пересечение, чем по словам;

– В качестве меры близости между документами выбран косинус. Близость меняется от 0 до 1.

3. Группировка документов, посвященных одним событиям, в кластеры и ведение сюжетной линии:

– Сюжетом являются упорядоченные во времени новостные сообщения, освещающие одно событие;

– Инструмент для формирования сюжетов – кластерный анализ;

– Вначале документы за определенный промежуток времени (предположим, 5 дней) объединяются в кластеры. Далее временное окно сдвигается на заданный временной интервал (допустим, 1 час). Кластеры обновляются: новые документы добавляются в кластеры, старые (не попадающие в сдвинутое временное окошко) – исключаются. Исключение старых документов из кластеров не означает их исключения из сюжета. Таким образом, сюжет – это текущие документы кластера и старые документы, которые раньше были в этом кластере;

– Сюжет заканчивается, когда в анализируемом наборе документов нет соответствующего ему кластера;

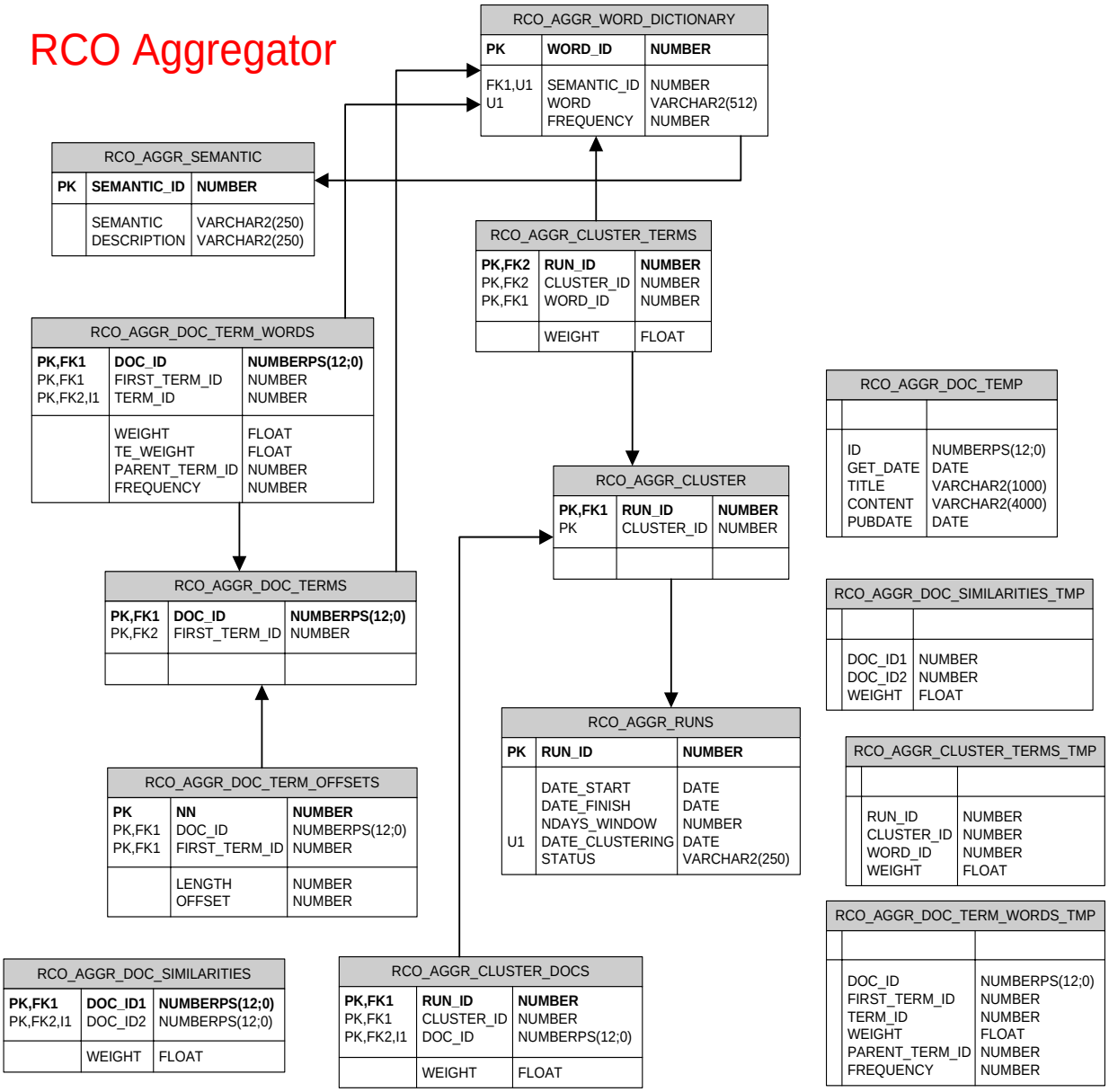
– Используется агломеративный иерархический кластерный анализ. Близость между документом и кластером равна средней близости данного документа со всеми документами кластера;

– Размер кластеров ограничивается введением порога. Число документов в кластере увеличивается, пока $min(Si) > T$, где T – наперед заданный порог, Si – среднее близости между i-м документом кластера и всеми остальными, минимум берется по всем документам кластера;

– Порог выбирается исходя из требований к полноте/точности ведения сюжетной линии.

Обработка данных

Структура данных агрегатора новостей



Взаимодействие с базой новостей

В базе данных новостей должны быть созданы таблицы, представления, пакеты и последовательность агрегатора новостей согласно предоставленному скрипту создания объектов.

Единственной точкой взаимодействия с данными, хранящимися в базе новостей, является список документов новостей в таблице **AGREGATED_NEWS**. Таблицы агрегатора ссылаются лишь на поле **AGREGATED_NEWS.**ID – агрегатор считывает данные из таблицы только на этапе лингвистического анализа документов.

При удалении новости из БД новостей в базе агрегатора каскадно будут удалены все записи, ссылающиеся на удаляемые новости. При удалении или изменении значительного количества новостных документов целесообразно пересчитать частоту встречаемости терминов и заново рассчитать кластеры за интервал времени, к которому принадлежали удаленные/измененные документы.

Для доступа в БД агрегатору необходим логин со следующими полномочиями и ограничениями:

- *Connect, resource* или квоты на *tablespace* (определяется политикой Заказчика);
- Чтение из таблицы **AGREGATED_NEWS**;
- Полный доступ к данным в таблицах, начинающихся с префикса *RCO_AGGR_*;
- Право вызывать функции, начинающиеся с **pkgrco_aggr**.

При разрыве соединения с БД программа не будет пытаться восстановить его, а завершится с критической ошибкой.

Ожидаемое потребление ресурсов

Объем таблиц агрегатора данной версии примерно в 3 раза превышает объем таблицы **AGREGATED_NEWS**. При необходимости реализации параллельных вычислений близости документов через таблицу **RCO_AGGR_DOC_SIMILARITIES** объем таблиц возрастет еще вдвое.

Поскольку каждая итерация кластеризации фиксируется в БД, специфических требований к журналам транзакций нет.

В файловой системе файл протокола агрегатора занимает примерно 10% от объема таблицы **AGREGATED_NEWS**.

Стратегия обработки ошибок

На время расчетов список новостных документов в базе данных не блокируется – всецело полагаемся на целостность ссылок.

Если возникнет ошибка, связанная с нарушением целостности БД агрегатора, выводится сообщение об ошибке в файл протокола. Критическая ошибка (требующая завершения программы) будет дополнительно зафиксирована в системном журнале. Все ошибки взаимодействия с БД считаются критическими.

В случае сбоя удаления/повреждения данных агрегатора необходимо произвести следующие операции (см. раздел «Операции по сопровождению, специфичные для БД агрегатора»):

- Остановить агрегатор;

- Определить охватывающий удаленные/поврежденные/некорректные данные интервал. Очистить данные агрегатора за указанный интервал;
- Пересчитать частоты встречаемости слов;
- Запустить агрегатор в ретроспективном режиме для получения корректных данных (если необходимо);
- Запустить агрегатор в штатном режиме.

Примечание. Если важно сохранение идентификаторов сходных кластеров во времени (т.е. должна прослеживаться сюжетная линия по набору документов и терминов, описывающих кластер, а кластер должен иметь единый числовой идентификатор во всех итерациях, где он существует), необходимо производить пересчет от нижней границы интервала со сбойными данными до текущего момента. При этом всегда должен быть включен параметр `WriteSimilarities=`*1*, а при запуске после сбоя нужно установить еще и параметр `ContinueLastIteration=`*1* для учета идентификаторов сформированных кластеров.

Описание некоторых ошибок

В файле протокола могут встретиться следующие записи об ошибках:

Ошибка анализа текста и выделения терминов из новости с указанным идентификатором (поле `ID` таблицы **`AGREGATED_NEWS`**), допустим:

```
[22/06/2007 05:00:54]: Exception while extracting terms from document
   47032 (file 'RCOAggregator.cpp' line 881)
[22/06/2007 05:00:54]: Error in core dll:. Location: line 515, file
   TermExtractor.cpp
```

Описание объектов базы данных агрегатора новостей

Элемент	Наименование	Тип данных	Описание
Таблица	**RCO_AGGR_CLUSTER**		**ИД кластера.**
Столбец	CLUSTER_ID	*NUMBER*	ИД кластера.
Столбец	RUN_ID	*NUMBER*	ИД итерации.
Таблица	**RCO_AGGR_CLUSTER_ DOCS**		**Документы кластера.**
Столбец	CLUSTER_ID	*NUMBER*	ИД кластера.
Столбец	DOC_ID	*NUMBER*	ИД документа.
Столбец	RUN_ID	*NUMBER*	ИД итерации.
Столбец	WEIGHT	*FLOAT*	Вес документа в кластере.
Таблица	**RCO_AGGR_CLUSTER_ TERMS**		**Термины – характеристики кластера.**
Столбец	CLUSTER_ID	*NUMBER*	ИД кластера.
Столбец	RUN_ID	*NUMBER*	ИД итерации.
Столбец	WEIGHT	*FLOAT*	Вес слова в кластере.
Столбец	WORD_ID	*NUMBER*	ИД слова или словосочетания.
Таблица	**RCO_AGGR_DOC_ SIMILARITIES**		**Вектор близостей документов (поскольку отношение симметрично, храним только вес для ИД1, ИД2, где ИД1 < ИД2, а для ИД2,ИД1 получим перестановкой столбцов).**
Столбец	DOC_ID1	*NUMBER*	ИД документа.
Столбец	DOC_ID2	*NUMBER*	ИД документа.
Столбец	WEIGHT	*FLOAT*	Относительный вес документов.
Таблица	**RCO_AGGR_DOC_ SIMILARITIES_TMP**		**Временная таблица для ускорения сохранения близостей документов.**
Столбец	DOC_ID1	*NUMBER*	ИД документа.
Столбец	DOC_ID2	*NUMBER*	ИД документа.
Столбец	WEIGHT	*FLOAT*	Относительный вес документов.
Таблица	**RCO_AGGR_DOC_TERMS**		**Термины документов. Каждый термин встречается в списке для документа лишь однажды.**
Столбец	DOC_ID	*NUMBER*	ИД документа.
Столбец	FIRST_TERM_ID	*NUMBER*	ИД первого (основного) слова или словосочетания.
Таблица	**RCO_AGGR_DOC_TERM_ OFFSETS**		**Смещение и длина для каждого вхождения термина в документ.**
Столбец	DOC_ID	*NUMBER*	ИД документа.
Столбец	FIRST_TERM_ID	*NUMBER*	ИД термина.
Столбец	LENGHT	*NUMBER*	Длина.
Столбец	NN	*NUMBER*	ID вхождения.
Столбец	OFFSET	*NUMBER*	Смещение.

Таблица	**RCO_AGGR_DOC_TERM_ WORDS**		**Список слов термина документа. Позже будет преобразован в дерево. Идентифицируется по первому слову списка.**
Столбец	DOC_ID	*NUMBER*	ИД документа.
Столбец	FIRST_TERM_ID	*NUMBER*	Идентификатор списка слов и словосочетаний термина документа (по первому слову списка).
Столбец	PARENT_TERM_ID	*NUMBER*	ИД родительского слова или словосочетания в дереве.
Столбец	TERM_ID	*NUMBER*	ИД слова или словосочетания, входящего в термин.
Столбец	FREQUENCY	*FLOAT*	Частота слова в документе.
Столбец	TE_WEIGHT	*FLOAT*	Вес слова, выдаваемые **TermExtrator'ом**.
Столбец	WEIGHT	*FLOAT*	Вес слова после перерасчета (*fTFiDFWeight*).
Таблица	**RCO_AGGR_RUNS**		**Итерации расчета кластеров.**
Столбец	DATE_FINISH	*DATE*	Дата окончания расчета.
Столбец	DATE_START	*DATE*	Дата запуска расчета.
Столбец	DATE_CLUSTERING	*DATE*	Время кластеризации (на какой момент времени сформирован данный набор кластеров).
Столбец	NDAYS_WINDOW	*NUMBER*	Ширина окна кластеризации в днях.
Столбец	RUN_ID	*NUMBER*	ИД итерации.
Столбец	STATUS	*VARCHAR2*	Состояние расчета (успех/ошибка).
Таблица	**RCO_AGGR_SEMANTIC**		**Семантика слов.**
Столбец	DESCRIPTION	*VARCHAR2*	Описание.
Столбец	SEMANTIC	*VARCHAR2*	Мнемоническое представление семантической информации (имена типов сущностей или частей речи).
Столбец	SEMANTIC_ID	*NUMBER*	Идентификатор.
Таблица	**RCO_AGGR_WORD_ DICTIONARY**		**Словарь слов и словосочетаний.**
Столбец	FREQUENCY	*NUMBER*	частота слова во всей коллекции документов (используется при вычислении вероятности появления слова).
Столбец	SEMANTIC_ID	*NUMBER*	Идентификатор.
Столбец	WORD	*VARCHAR2*	Само слово или словосочетание.
Столбец	WORD_ID	*NUMBER*	ИД слова или словосочетания.

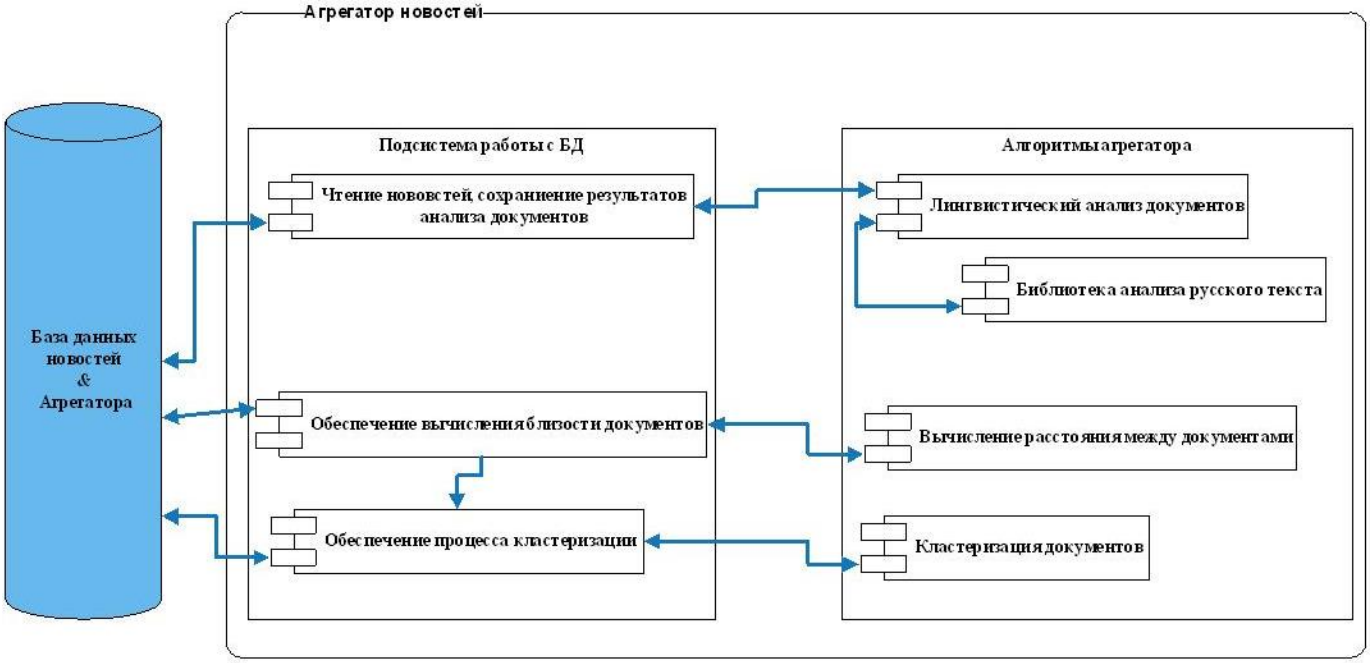
Для работы с результатами последней итерации кластеризации существует набор представлений:

Представление (View)	**Описание**
V_RCO_AGGR_CLUSTER	Текущий (последний по времени) набор кластеров.
V_RCO_AGGR_CLUSTER_DOCS	Документы текущего набора кластеров.
V_RCO_AGGR_CLUSTER_OLD_DOCS	Документы, вышедшие из текущего набора кластеров.
V_RCO_AGGR_CLUSTER_TERMS	Термины текущего набора кластеров.

Процедуры и функции	
procedure pkgrco_aggr_aggreagator_reset (dtfrom date, dtto date, bClearDictionary boolean:= false);	Удаляет данные о терминах документов, кластерах, запусках за указанный интервал из таблиц с префиксом *RCO_AGGR_* за исключением словарей слов и семантики. Если установлен параметр `bClearDictionary` – удаляет неиспользуемые слова из словаря.
procedure pkgrco_aggr_cluster_describing_terms (nRunID integer, treshold integer:= null);	Вычисляет *treshold* наиболее характерных терминов для кластера с *nClusterID* на итерации *nRunID* и сохраняет их (реализацию скопировать 1 в 1 с аналогичной **C++** функции стенда; сохранять 10 терминов с максимальным весом).
function pkgrco_aggr_get_last_run_id return integer;	Выдает идентификатор последнего (по времени итерации кластеризации) успешного запуска агрегатора.
pkgrco_aggr_fnc_rco_insert_semantic pkgrco_aggr_fnc_rco_insert_word	Функции для пополнения словаря семантики и значений слов.
Procedure pkgrco_aggr_term_update_frequencies;	Для всех записей словаря обновляет частоту встречаемости.
function pkgrco_aggr_get_next_id return integer;	Возвращает следующий идентификатор последовательности *RCO_AGGR_SEQUENCE* (в перспективе – сделать версию, выдающую заданное количество идентификаторов).
Procedure pkgrco_aggr_prc_RCO_SYNC_RUBRICS_4_CLUSTER	Для заданной итерации ставит нерубрицированные документы из ее кластеров в очередь Рубрикатора.

Взаимодействие БД агрегатора и алгоритмов агрегатора

Для каждого алгоритма агрегатора реализуем соответствующую ему задачу обеспечения чтения входных данных из БД и сохранения выходных данных в БД. На приведенной ниже диаграмме показаны направления обмена данным между задачами агрегатора:



Функции систем, изображенных на диаграмме:

Алгоритмы агрегатора выполняют собственно анализ и кластеризацию.

База данных хранит входной новостной поток и результаты кластеризации. Структура БД описана в предыдущей секции.

Подсистема работы с БД обеспечивает:

- инициализацию (с разбором параметров), деинициализацию и завершение приложения, а также подключение к/отсоединение от БД агрегатора;
- обмен данными между алгоритмами агрегатора и БД и управляет границами временных отрезков (сдвигает окно итерации, выравнивает время итерации);
- организацию цикла непрерывных вычислений в ретроспективном режиме и цикл с ожиданием наступления времени следующей итерации в штатном режиме;
- обработку исключения алгоритмов агрегатора и ошибки/исключения БД;
- протоколирование работы программы.

Замечания по реализации

1. Особенности поведения системы в штатном режиме:

   – Время итерации будет выровнено по границе часа, если не указан параметр `noalign`. При запуске агрегатора без указания параметра `noalign` в штатном режиме первая итерация будет сделана в начале часа, когда был произведен запуск;

   – Если за время очередного шага кластеризации в базу не поступило ни одного документа, кластеризация не производится, однако часть «старых» документов выходит из рассмотрения;

   – Если время расчета итерации кластеризации больше шага кластеризации, следующие одна или несколько итераций будут пропущены с выдачей предупреждения в файл протокола.

2. Особенности поведения системы в ретроспективном режиме:

   – Если за время очередного шага кластеризации в базе не найдено ни одного «нового» (не участвовавшего в расчете в данном прогоне) документа, переходим к следующему шагу, и так до тех пор, пока не окно кластеризации не сдвинется настолько, чтобы в него попали «новые» документы;

   – Время итерации всегда выровнено по границе часа;

   – На каждой итерации (в том числе и когда просто сдвигается окно из-за отсутствия документов) удаляются результаты кластеризации предыдущих прогонов, попадающих в интервал (время_итерации - шаг_кластеризации, время_итерации].

3. При первой итерации кластеризации, независимо от режима, пытаемся найти старейший результат кластеризации за интервал времени [время_кластеризации - шаг_кластеризации, время_кластеризации) и использовать его в качестве первоначального набора кластеров для сохранения преемственности идентификаторов похожих кластеров.

4. В один момент времени в штатном режиме имеет смысл запускать только один экземпляр агрегатора. Предполагается обеспечивать монопольность штатной кластеризации путем ограничения количества сессий на логин агрегатора.

5. Задача лингвистического анализа документов с выделением ключевых терминов (режим **extract_only**) допускает параллельное исполнение нескольких экземпляров. Пользователь должен задать непересекающиеся интервалы кластеризации для задач.

6. Задача ретроспективного расчета кластеров (параметр `interval`) допускает параллельное исполнение нескольких экземпляров. Пользователь должен задать непересекающиеся интервалы кластеризации для задач.

7. Для обеспечения повторяемости времен итераций кластеризации имеет смысл задавать значение шага кластеризации, которое делит количество часов в сутках нацело.

Операции по сопровождению, специфичные для БД агрегатора

Удаление данных агрегатора

Удаление всех данных агрегатора за указанный интервал производится вызовом `PKGRCO_AGGR_AGGREAGATOR_RESET` с последующей фиксацией транзакции. Если задать параметр `bClearDictionary`, будут удалены неиспользуемые слова из словаря слов (**`RCO_AGGR_WORD_DICTIONARY`**). Словарь семантических признаков (**`RCO_AGGR_SEMANTIC`**) не очищается.

Задача уточнения частоты встречаемости

При массовом удалении документов новостей и связанных с ними данных агрегатора необходимо пересчитать частоты встречаемости терминов в тексте – поле `FREQUENCY` таблицы **`RCO_AGGR_WORD_DICTIONARY`**. Расчет производится вызовом `PKGRCO_AGGR_TERM_UPDATE_FREQUENCIES` с последующей фиксацией транзакции.

Ретроспективный расчет кластеров

Для запуска агрегатора в ретроспективном режиме необходимо задать параметр `interval`**.** Перед повторным запуском ретроспективного расчета на интервал, где уже есть рассчитанные данные, необходимо выполнить две вышеописанные операции по сопровождению.

Переход на летнее/зимнее время

Система, работающая в штатном режиме, должна быть остановлена перед моментом перехода потока новостей на летнее/зимнее время и после запущена вновь (зависимость от времени сервера БД).