Абдышова А.Т.

СЕМАНТИЧЕСКИЙ ПОИСК НА ФАЙЛАХ С ИСПОЛЬЗОВАНИЕМ SQL SERVER

A.T. Abdyshova

SEMANTIC SEARCH ON THE FILES USING SQL SERVERD

УДК:624.012:801.52

Для более глубокого анализа неструктурированных текстов с использованием SQL Server.

For more deep analyze SQL Server used for not structural texts.

Есть несколько методов для реализации семантического поиска \4\. Неструктурированные данные, это текст, который не содержит машиночитаемый семантическую информацию. Лучший способ поиска или индексации зависит от структуры данных. Один из типов структурированных данных являются онтологии. Онтология формально описывает имеющиеся концепции в конкретной области. Один из возможностью запрашивать структурированных данных является концептуальный соответствующий график \1-5\. В этом методе каждый запрос и данные представлены в виде деревьев понятий (онтологии). Поиск сравнивает запрос с каждого дерева в базе данных и находит наиболее подходящий дерево.

Один из способов, чтобы представить такую структуру данных является использование RDF (Resource Description Frame work) это конструкция, используемая для описания данных. Он может быть использован для описания структуры данных так же, как модели класса домен. Используя эту сеть соединительных концепций, поисковая система понимает контекст поиска и может получить более точные результаты поиска. Используя эту концептуальную сеть можно также сделать смысл слово неоднозначности \8\. Если пользователь ищет слово с несколькими значениями, поисковая система выбирает наиболее вероятное значение, исследуя другие слова в запросе и доступные понятия.

Проблема выше методов является то, что они требуют структурированные данные - они не подходят для доступа неструктурированные данные, такие как текстовые документы. Во многих распространенных есть и много текстовых документов, таких как веб-страниц, файлов слов или текстовых столбцов в базе данных. Поскольку Microsoft SQL Server не имеет метаданные. Чтобы запросить неструктурированных данных, поисковая система должна индексировать все документы, разбить его на ключевых слов и забить их в зависимости от статистического анализа. Большинство поисковых систем индексировать все документы с извлечения терминологии или алгоритма извлечения фраза. Следующий шаг заключается в удалении шумовые слова. Это слова, которые слишком часто и которые не содержат полезной информации. Некоторые примеры слова "в" или "". Эти слова также называют стоп-слова. Обычно поисковая система имеет стоп базу данных слов каждого языка. Некоторые поисковые системы не удаляют стоп-слова для более эффективной поддержки точная фраза.

Еще одна важная вещь в семантического поиска является перегиба и вытекающие. Синонимы является модификация слова в другую форму - например другой напряженной, дело и т.д. Преобразование слова обратно его основа называется вытекающие. Использование морфологических форм слов в поисковой системе использует только стебли из слов и, следовательно, не может сравниться слова, даже если пользователь выполняет поиск со словами в неправильной формы. Существует, например, прошедшего времени, вытекающие который используется, чтобы помочь глаголам в различных сопряжений.

Для использования семантического поиска непосредственно на файлах в файловой системе у нас есть создать новый поток, файл включают таблицу также как таблицы файлов\7-10\:

CREATEDATABASEMyDatabaseONPRIMARY
(NAME=MyDatabaseDB,FILENAME='C:\MyDatabase\MyDatabaseDB.mdf'),

FILEGROUPMyDatabaseFSCONTAINSFILESTREAM(NAME=MyFSDatabaseFS,FILENAME='C:\MyDatabase\MyDatabaseFS')

LOGON (NAME=MyFSDatabaseLOG,FILENAME='C:\MyDatabase\MyDatabaseLOG.ldf')

После того, как база данных была создана, у нас есть для того, чтобы не-транзакционный доступ на уровне базы данных:

ALTERDATABASEMyDatabaseSETFILESTREAM
(Non_Transacted_Access=FULL,Directory_name=N'MyDatabase')

Теперь мы можем «преобразовать» таблицу в таблицу файла с помощью следующего запроса:

USEMyDatabaseCREATETABLEMyFileTableASFILETABLEWITH (FileTable_Directory=N'MyFiles')

Использование следующий запрос мы можем увидеть список трёх файлов:1) термины_механики, 2) термины_информатика, 3) термины_иностранных языков в файловой таблице №1: SELECT* FROMdbo. My File Table

```
SELECT * FROM dbo.MyFileTable

create unique index MyPK on MyFileTable(stream_id)

CREATE FULLTEXT CATALOG MyFileTableCatalog WITH ACCENT_SENSITIVITY = ON;
CREATE FULLTEXT INDEX ON MyFileTable
  ( name LANGUAGE 1049 STATISTICAL_SEMANTICS, file_type LANGUAGE 1049 STATISTICAL_SEMANTICS,
  file_stream TYPE COLUMN file_type LANGUAGE 1049 STATISTICAL_SEMANTICS )
  KEY INDEX MyPK ON MyFileTableCatalog WITH CHANGE_TRACKING AUTO, STOPLIST = SYSTEM;
```

Результаты | Сообщения

stream_id	file_stream	name	path_locator	parent_pat...	file_...	cache...	creation_time
9B74E56E-92E8-E311-9883-90004EA9A4C6	0xD0CF11E0A1B11AE10000...	термины.doc.doc	0xFE161743CE813B4FE5901460FD309...	NULL	doc	45153...	2014-05-31 13:08
6D107FCF-95E8-E311-9883-90004EA9A4C6	0xD0CF11E0A1B11AE10000...	Информатика-4.досновый...	0xFEECF0F967D6238FEE1256073D6B...	NULL	doc	45153...	2014-05-31 13:33
6F107FCF-95E8-E311-9883-90004EA9A4C6	0xD0CF11E0A1B11AE10000...	с__зд__к иностр.doc	0xFD2C4099DA1811EFCF51768699CA...	NULL	doc	10726...	2014-05-31 13:33

таблица №1

После создания базы данных и таблицы файлов мы должны установить семантический поисковый индекс на таблице файлов. Следующий запрос создает индекс. Первичный ключ ("MyPK" в списке) должен быть заменен с первичного ключа именем из ранее созданной таблицы файлов. При необходимости заменить язык с нужным кодом языка. Важно соответствовать языку индекса с языками в документах.

Createuniqueindex My PK on My File Table (stream id)

CREATEFULLTEXTCATALOG My File Table CatalogWITHACCENT_SENSITIVITY=ON;

CREATEFULLTEXTINDEXON My FileTable (name LANGUAGE 1049 STATISTICAL SEMANTICS, file_type LANGUAGE 1049 STATISTICAL SEMANTICS, file_stream TYPECOLUMN file_type LANGUAGE 1049 STATISTICAL_SEMANTICS)

KEYINDEX My PKON My FileTable Catalog WITHCHANGE_TRACKINGAUTO,STOPLIST = SYSTEM;

Чтобы проверить созданный индекс, мы скопировать некоторые файлы в Windows, доля потока файлов и подождать некоторое время, пока индекс не был построен.

Следующий запрос показывает все ключевые слова из всех документов, заказанных их оценки в таблице №2:

SELECT*FROMdbo.MyFileTable

SELECTname,document_key,keyphrase,scoreFROMsemantickeyphrasetable(MyFileTable,*)

INNERJOINMyFileTableONstream_id=document_keyORDERBYname,scoreDESC

```
SELECT name, document_key, keyphrase, score FROM semantickeyphrasetable(MyFileTable, *)
INNER JOIN MyFileTable ON stream_id = document_key ORDER BY name, score DESC

SELECT * FROM semantickeyphrasetable(MyFileTable, *) ORDER BY score DESC

SELECT stream_ID, Name, keyphrase, score FROM semantickeyphrasetable(MyFileTable, *) INNER JOIN MyFileTable ON stream_ID = document_key ORDER

SELECT Name FROM semantickeyphrasetable (MyFileTable, *) INNER JOIN MyFileTable ON stream_ID = document_key WHERE keyphrase = 'проекция'

SELECT COUNT(*) AS 'Number of SQL documents' FROM semantickeyphrasetable (MyFileTable, *) WHERE keyphrase = 'физ.-мат.наук'
```

Результаты | Сообщения

name	document_key	keyphrase	score
Информатика-4.досновый1.doc	6D107FCF-95E8-E311-9883-90004EA9A4C6	сан	0,5425657
Информатика-4.досновый1.doc	6D107FCF-95E8-E311-9883-90004EA9A4C6	тамыр	0,5238146
Информатика-4.досновый1.doc	6D107FCF-95E8-E311-9883-90004EA9A4C6	жана	0,5195129
Информатика-4.досновый1.doc	6D107FCF-95E8-E311-9883-90004EA9A4C6	белги	0,5118694
Информатика-4.досновый1.doc	6D107FCF-95E8-E311-9883-90004EA9A4C6	бет	0,5071914
Информатика-4.досновый1.doc	6D107FCF-95E8-E311-9883-90004EA9A4C6	алгебра	0,5065572
Информатика-4.досновый1.doc	6D107FCF-95E8-E311-9883-90004EA9A4C6	геометрия	0,5022383
Информатика-4.досновый1.doc	6D107FCF-95E8-E311-9883-90004EA9A4C6	уравнение	0,5012182
Информатика-4.досновый1.doc	6D107FCF-95E8-E311-9883-90004EA9A4C6	геометрическое	0,4905783
Информатика-4.досновый1.doc	6D107FCF-95E8-E311-9883-90004EA9A4C6	симметрия	0,4778332
Информатика-4.досновый1.doc	6D107FCF-95E8-E311-9883-90004EA9A4C6	теорема	0,4771576

таблица №2

Запросы. Семантическая функция поиска в Microsoft SQL Server 2012 позволяет создавать три типа запросов:

1.Ключевые фразы: Двигатель позволяет получить доступ к статистически значимых фраз в каждом документе.

2. Подобные документы: Этот тип запроса поможет найти аналогичные или связанные документы или строки на основе ключевых фраз, содержащихся в документах.

3.Сходство объяснение: Этот запрос возвращает ключевые фразы, которые объясняют, почему два ряда или документы были определены как близкие.

Найти ключевые фразы в документе (semantickeyphrasetable).Использование семантический поисковый индекс можно получить все проиндексированные ключевые слова и их оценка в таблице №3:

SELECT*FROM semantickey phrase table (My File Table,*) ORDERBY score DESC

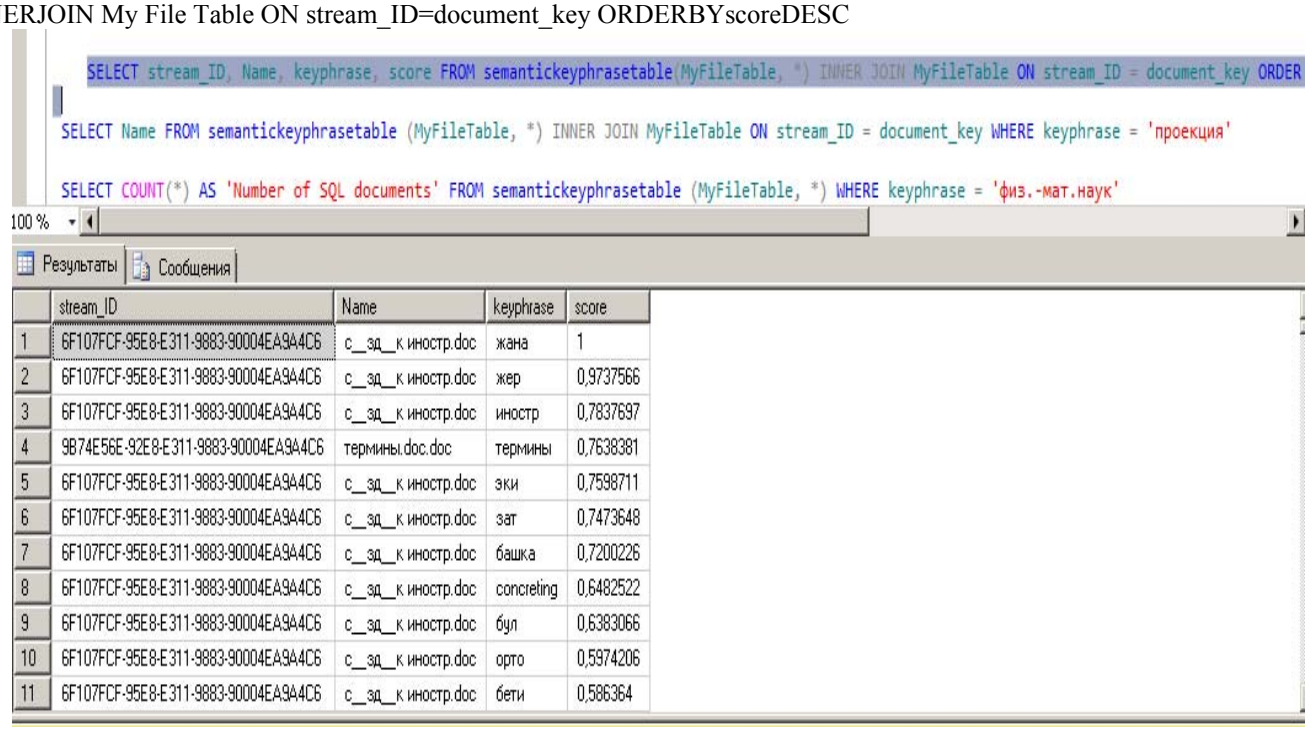
```
SELECT * FROM semantickeyphrasetable(MyFileTable, *) ORDER BY score DESC

SELECT stream_ID, Name, keyphrase, score FROM semantickeyphrasetable(MyFileTable, *) INNER JOIN MyFileTable ON stream_ID = document_key ORDER

SELECT Name FROM semantickeyphrasetable (MyFileTable, *) INNER JOIN MyFileTable ON stream_ID = document_key WHERE keyphrase = 'проекция'

SELECT COUNT(*) AS 'Number of SQL documents' FROM semantickeyphrasetable (MyFileTable, *) WHERE keyphrase = 'физ.-мат.наук'
```

Результаты | Сообщения

	column_id	document_key	keyphrase	score
1	2	6F107FCF-95E8-E311-9883-90004EA9A4C6	жана	1
2	2	6F107FCF-95E8-E311-9883-90004EA9A4C6	жер	0,9737566
3	3	6F107FCF-95E8-E311-9883-90004EA9A4C6	иностр	0,7837697
4	3	9B74E56E-92E8-E311-9883-90004EA9A4C6	термины	0,7638381
5	2	6F107FCF-95E8-E311-9883-90004EA9A4C6	эки	0,7598711
6	2	6F107FCF-95E8-E311-9883-90004EA9A4C6	зат	0,7473648
7	2	6F107FCF-95E8-E311-9883-90004EA9A4C6	башка	0,7200226
8	2	6F107FCF-95E8-E311-9883-90004EA9A4C6	concreting	0,6482522
9	2	6F107FCF-95E8-E311-9883-90004EA9A4C6	бул	0,6383066
10	2	6F107FCF-95E8-E311-9883-90004EA9A4C6	орто	0,5974206
11	2	6F107FCF-95E8-E311-9883-90004EA9A4C6	бети	0,586364

таблица №3

Использование SQL присоединиться мы можем добавить имя документа в результате ,как показана в таблице №4: SELECT stream_ID, Name, keyphrase, score FROM semantickey phrase table (My File Table,*) INNERJOIN My File Table ON stream_ID=document_key ORDERBYscoreDESC



```
SELECT stream_ID, Name, keyphrase, score FROM semantickeyphrasetable(MyFileTable, *) INNER JOIN MyFileTable ON stream_ID = document_key ORDER

SELECT Name FROM semantickeyphrasetable (MyFileTable, *) INNER JOIN MyFileTable ON stream_ID = document_key WHERE keyphrase = 'проекция'

SELECT COUNT(*) AS 'Number of SQL documents' FROM semantickeyphrasetable (MyFileTable, *) WHERE keyphrase = 'физ.-мат.наук'
```

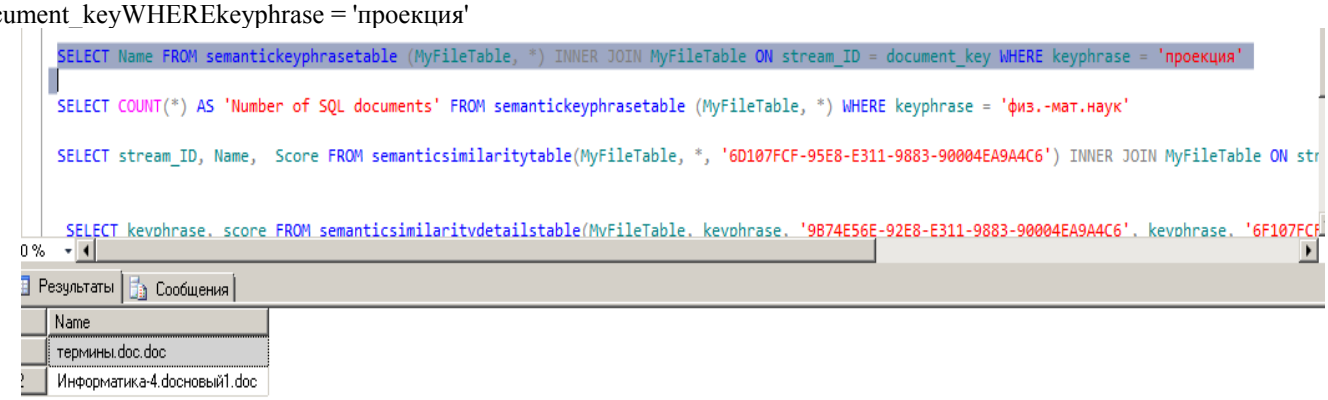
100 % — Результаты | Сообщения

	stream_ID	Name	keyphrase	score
1	6F107FCF-95E8-E311-9883-90004EA9A4C6	с__зд__к иностр.doc	жана	1
2	6F107FCF-95E8-E311-9883-90004EA9A4C6	с__зд__к иностр.doc	жер	0,9737566
3	6F107FCF-95E8-E311-9883-90004EA9A4C6	с__зд__к иностр.doc	иностр	0,7837697
4	9B74E56E-92E8-E311-9883-90004EA9A4C6	термины.doc.doc	термины	0,7638381
5	6F107FCF-95E8-E311-9883-90004EA9A4C6	с__зд__к иностр.doc	эки	0,7598711
6	6F107FCF-95E8-E311-9883-90004EA9A4C6	с__зд__к иностр.doc	зат	0,7473648
7	6F107FCF-95E8-E311-9883-90004EA9A4C6	с__зд__к иностр.doc	башка	0,7200226
8	6F107FCF-95E8-E311-9883-90004EA9A4C6	с__зд__к иностр.doc	concreting	0,6482522
9	6F107FCF-95E8-E311-9883-90004EA9A4C6	с__зд__к иностр.doc	бул	0,6383066
10	6F107FCF-95E8-E311-9883-90004EA9A4C6	с__зд__к иностр.doc	орто	0,5974206
11	6F107FCF-95E8-E311-9883-90004EA9A4C6	с__зд__к иностр.doc	бети	0,586364

таблица №4

С помощью следующего запроса мы можем получить список всех документов, которые содержат ключевое слово " проекция " в таблице№ 5:

SELECT Name FROM semantickey phrase table (My File Table,*) INNERJOIN My File Table ON stream_ID = document_keyWHEREkeyphrase = 'проекция'



```
SELECT Name FROM semantickeyphrasetable (MyFileTable, *) INNER JOIN MyFileTable ON stream_ID = document_key WHERE keyphrase = 'проекция'

SELECT COUNT(*) AS 'Number of SQL documents' FROM semantickeyphrasetable (MyFileTable, *) WHERE keyphrase = 'физ.-мат.наук'

SELECT stream_ID, Name,  Score FROM semanticsimilaritytable(MyFileTable, *, '6D107FCF-95E8-E311-9883-90004EA9A4C6') INNER JOIN MyFileTable ON str

SELECT keyphrase, score FROM semanticsimilaritydetailstable(MyFileTable, keyphrase, '9B74E56E-92E8-E311-9883-90004EA9A4C6', keyphrase, '6F107FCF
```

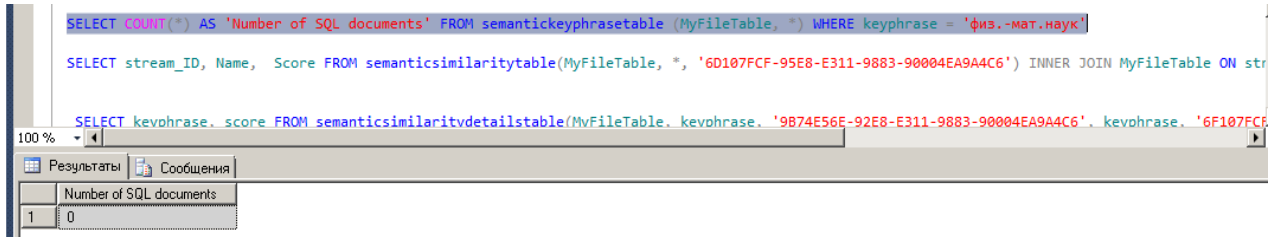
0 % — Результаты | Сообщения

	Name
	термины.doc.doc
2	Информатика-4.досновый1.doc

таблица №5

Другим примером является следующий запрос, который подсчитывает количество документов с конкретным ключевым словом в таблице №6:

SELECTCOUNT (*) AS' Number of SQL documents' FROM semantickey phrase table (My File Table,*) WHEREkey phrase='физ.-мат.наук'



```
SELECT COUNT(*) AS 'Number of SQL documents' FROM semantickeyphrasetable (MyFileTable, *) WHERE keyphrase = 'физ.-мат.наук'

SELECT stream_ID, Name,  Score FROM semanticsimilaritytable(MyFileTable, *, '6D107FCF-95E8-E311-9883-90004EA9A4C6') INNER JOIN MyFileTable ON str

SELECT keyphrase, score FROM semanticsimilaritydetailstable(MyFileTable, keyphrase, '9B74E56E-92E8-E311-9883-90004EA9A4C6', keyphrase, '6F107FCF
```

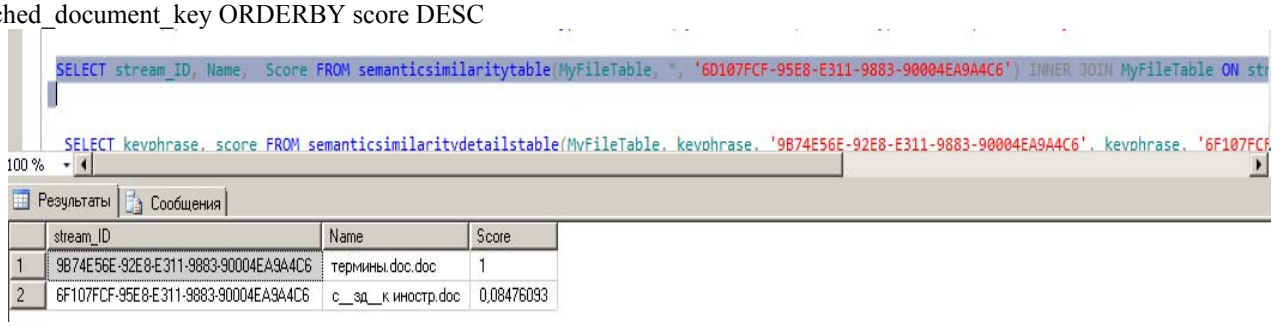
100 % — Результаты | Сообщения

	Number of SQL documents
1	0

таблица №6

Найти похожие или связанные документы (semantic simil arity table).

С процедурой semantic simi larity table мы можем получить список похожих документов для данного документа в таблице №7. SELECTstream_ID, Name, Score FROM semantic similaritytable (My File Table,*,' 6D107FCF-95E8-E311-9883-90004 EA9A4C6') IN NERJOIN My File Table ON stream_ID = matched_document_key ORDERBY score DESC



	stream_ID	Name	Score
1	9B74E56E-92E8-E311-9883-90004EA9A4C6	термины.doc.doc	1
2	6F107FCF-95E8-E311-9883-90004EA9A4C6	с__зд__к иностр.doc	0,08476093

таблица №7

Процедура semanticsimilaritydetailstable извлекает ключевые слова, которые делают два документа похожи. SELECT keyphrase, score FROM semantic simi larity details table(My File Table, keyphrase,'9B74E56E-92E8-E311-9883-90004 EA 9A4C6', keyphrase, '6F107FCF-95E8-E311-9883-90004EA9A4C6')

Заключение

Установка и использование Microsoft SQL Server 2012 семантический поиск прост. Согласно Microsoft \6\ индекс полнотекстового весы свыше 100 млн. документов и, таким образом может быть использован с очень большими наборами документов. Текущая реализация главной цели Семантический поиск является, чтобы найти похожие документы для данного документа. Эта функция работает удовлитворительно, однако, нужно определить собственные стоп-слова, чтобы отфильтровать нежелательные слова. Одно из преимуществ функции является то, что он работает на обычных текстовых столбцов, а также документов, хранящихся в файловой системе.

На данный момент, Microsoft SQL Server 2012 поддерживает только юниграмм. Это ограничивает использование в простого сравнения ключевых слов семантического интеллекта. Кроме того использования семантического поиска очень ограничены, только для поиска похожих документов для данного документа. На самом деле очень много механизмов работают неправильно (например, вытекающих) - другие просто не реализованы (например, фразы из нескольких слов).

В моих глазах, семантический поиск только небольшое расширение для полнотекстового поиска. Чтобы действительно получить большую ценность мы должны иметь возможность для поиска целых предложений .

Список использованных литератур:

1. Y. Chen, E. K. Garcia and R. M. Gupta, "Similarity-based Classification: Concepts and Algorithms".
2. Online. Available: http://www.google.com/. Accessed 22 1 2013
3. J. Ramos, "Using TF-IDF to Determine Word Relevance in Document Queries".
4. Online. Available: http://www.seco.tkk.fi/publications/2005/makela-semantic-search-2005.pdf
5. Online. Available: http://yury.name/algoweb/08algoweb-hand.pdf. Accessed 22 1 2013
6. Online. Available: http://msdn.microsoft.com/en-us/library/cc280405.aspx. Accessed 22 1 2013.
7. R. Mistry and S. Misner, Introducing Microsoft®SQL Server2012, Microsoft Press.
8. Online. Available: http://en.wikipedia.org/wiki/Semantic_search. Accessed 22 1 2013.
9. М.Д. Кутуев, А.Т. Абдышова "Методы для семантического отображения технической информации в моделях "
10. М.Д. Кутуев, А.Т. Абдышова Анализ методов семантической обработки текстов и повышение эффективности текстовой информации.

Рецензент: д.ф-м.н. Иманалиев Т.М.