Основы текстовой аналитики в Power Query

Это фрагмент книги Гил Равив. Power Query в Excel и Power BI: сбор, объединение и преобразование данных.

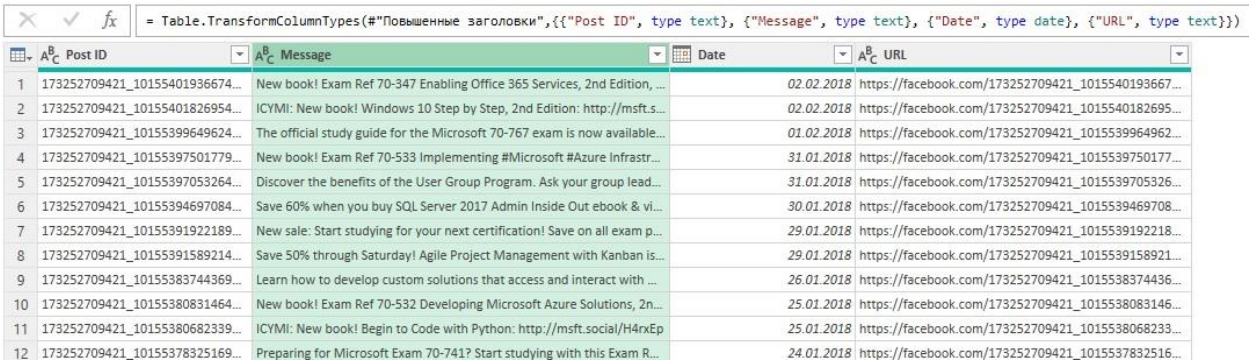
Предыдущий раздел К содержанию Следующий раздел

Текстовые столбцы позволяют извлекать дополнительный контекст. Описанные ниже методы полезны при анализе отзывов клиентов, комментариев или социальных сетей.

Поиск ключевых слов в текстовых столбцах

В следующем упражнении мы будем искать ключевые слова в сообщениях, которые Microsoft Press использует в постах на своей официальной странице в Facebook. Допустим, вам нужно оценить, какие из тем наиболее популярны в обсуждениях на Facebook: Microsoft Excel, Visual Studio, Azure, Windows.

Загрузите файл C11E01.xlsx и сохранить его в папке C:\Data\C11\. Откройте новую книгу Excel. Пройдите *Данные –> Получить данные –> Из файла –> Из книги*. Выберите файл C11E01.xslx и щелкните *Импорт*. В окне *Навигатор* выберите *Sheet1* и щелкните *Преобразовать данные*. Обратите внимание, что посты Microsoft Press Facebook сохраняются в столбце *Message*. Ваша задача – выполнить поиск ключевых слов в этом столбце.



```
= Table.TransformColumnTypes(#"Повышенные заголовки",{{"Post ID", type text}, {"Message", type text}, {"Date", type date}, {"URL", type text}})
```

	Post ID	Message	Date	URL
1	173252709421_10155401936674...	New book! Exam Ref 70-347 Enabling Office 365 Services, 2nd Edition, ...	02.02.2018	https://facebook.com/173252709421_10155401936674...
2	173252709421_10155401826954...	ICYMI: New book! Windows 10 Step by Step, 2nd Edition: http://msft.s...	02.02.2018	https://facebook.com/173252709421_10155401826954...
3	173252709421_10155399649624...	The official study guide for the Microsoft 70-767 exam is now available...	01.02.2018	https://facebook.com/173252709421_10155399649624...
4	173252709421_10155397501779...	New book! Exam Ref 70-533 Implementing #Microsoft #Azure Infrastr...	31.01.2018	https://facebook.com/173252709421_10155397501779...
5	173252709421_10155397053264...	Discover the benefits of the User Group Program. Ask your group lead...	31.01.2018	https://facebook.com/173252709421_10155397053264...
6	173252709421_10155394697084...	Save 60% when you buy SQL Server 2017 Admin Inside Out ebook & vi...	30.01.2018	https://facebook.com/173252709421_10155394697084...
7	173252709421_10155391922189...	New sale: Start studying for your next certification! Save on all exam p...	29.01.2018	https://facebook.com/173252709421_10155391922189...
8	173252709421_10155391589214...	Save 50% through Saturday! Agile Project Management with Kanban is...	29.01.2018	https://facebook.com/173252709421_10155391589214...
9	173252709421_10155383744369...	Learn how to develop custom solutions that access and interact with ...	26.01.2018	https://facebook.com/173252709421_10155383744369...
10	173252709421_10155380831464...	New book! Exam Ref 70-532 Developing Microsoft Azure Solutions, 2n...	25.01.2018	https://facebook.com/173252709421_10155380831464...
11	173252709421_10155380682339...	ICYMI: New book! Begin to Code with Python: http://msft.social/H4rxEp	25.01.2018	https://facebook.com/173252709421_10155380682339...
12	173252709421_10155378325169...	Preparing for Microsoft Exam 70-741? Start studying with this Exam R...	24.01.2018	https://facebook.com/173252709421_10155378325169...

Рис. 1. Сообщения Microsoft Press на Facebook в редакторе Power Query

Поскольку Power Query при поиске чувствителен к регистру, продублируем столбец *Message* и переведем все символы в нижний регистр. Исходные сообщения сохранятся без изменений, что важно, если следует включить исходный текст в отчет и сохранить некоторые символы в сообщениях в верхнем регистре.

Выделите столбец *Message.* Пройдите *Добавить столбец –> Создать дубликат столбца*. Выделите столбец *Копия Message* и пройдите *Преобразование –> Формат –> нижний регистр*. Переименуйте столбец *Копия Message* в *нижний регистр.*

Теперь создадим столбец *Тема* и заполним его одной из обнаруженных тем: Microsoft Excel, Visual Studio, Azure или Windows. Пройдите *Добавить столбец –> Условный столбец*. Настройте окно *Добавление условного столбца*:

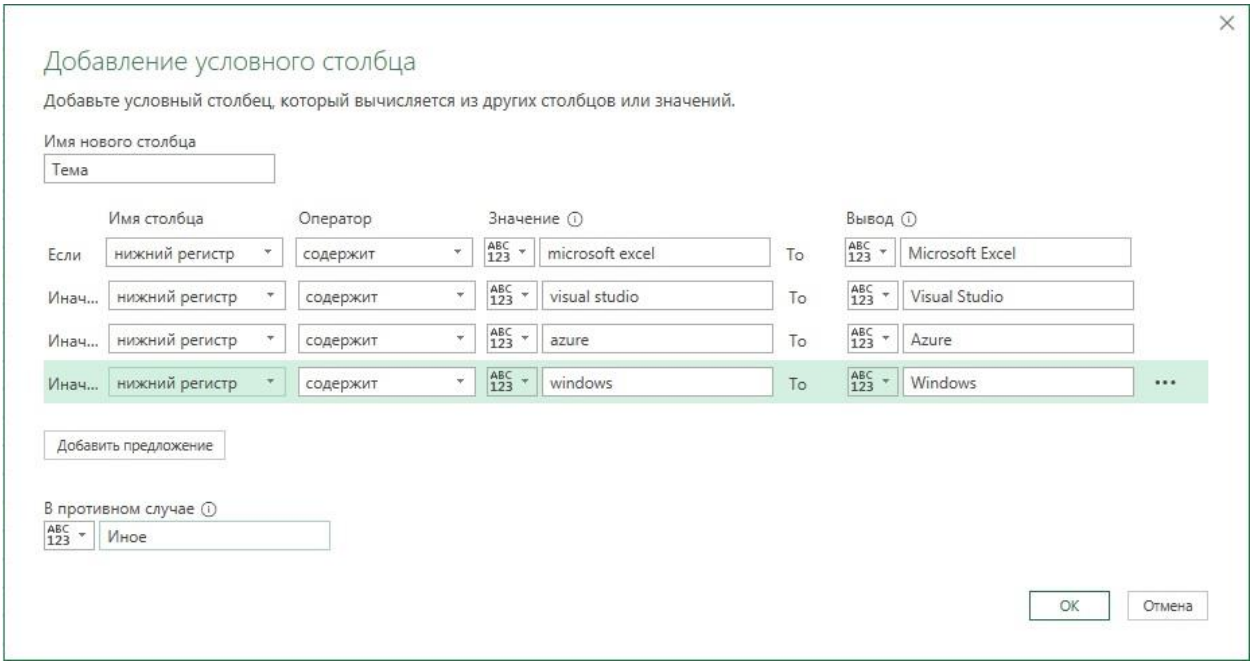


Рис. 2. Окно *Добавление условного столбца*

Щелкните мышью на элементе управления фильтром в заголовке столбца *Тема*, отмените выбор *Иное*. Удалите столбец *нижний регистр*, переименуйте запрос в *Microsoft Press Posts* и загрузите в таблицу на лист Excel. Файл решения C11E01 - Solution.xlsx.

Примененный метод обладает рядом особенностей:

1. Если ключевых слов много, их ручное добавление в окне *Добавление условного столбца* будет напрягать. Желательно найти альтернативный способ.
2. Ключевые слова жестко закодированы в формуле. Если необходимо загрузить список ключевых слов из внешнего источника, следует найти метод, который позволит динамически управлять ключевыми словами.
3. Использован принцип: одно сообщение – одна тема. Некоторые посты могут содержать несколько ключевых слов. Лучше учитывать каждое ключевое слово.
4. Ключевые слова, содержащиеся в качестве подстрок более длинных слов, могут быть указаны ошибочно. Лучше разделить сообщения на отдельные слова и уже их сопоставить с ключевыми словами.

Применение декартова произведения для определения ключевых слов

Для масштабирования решения, примененного выше, и обнаружения большого числа тем из динамического списка ключевых слов можно выполнять декартово произведение между постами Microsoft Press и списком ключевых слов. При этом формируется временная таблица большего размера. Таблица состоит из всех комбинаций между постами в Facebook и ключевыми словами.

> В теории множеств декартово произведение может применяться к нескольким наборам, что позволит сформировать перечень всех комбинаций элементов из разных наборов — по одному из каждого набора.

Сопоставляя все комбинации между несколькими наборами данных или объектами, можно осуществить вычисления «что если», выполнить анализ рыночной корзины и реализовать вычисления с помощью методов из теории графов (например, поиск кратчайшего пути).

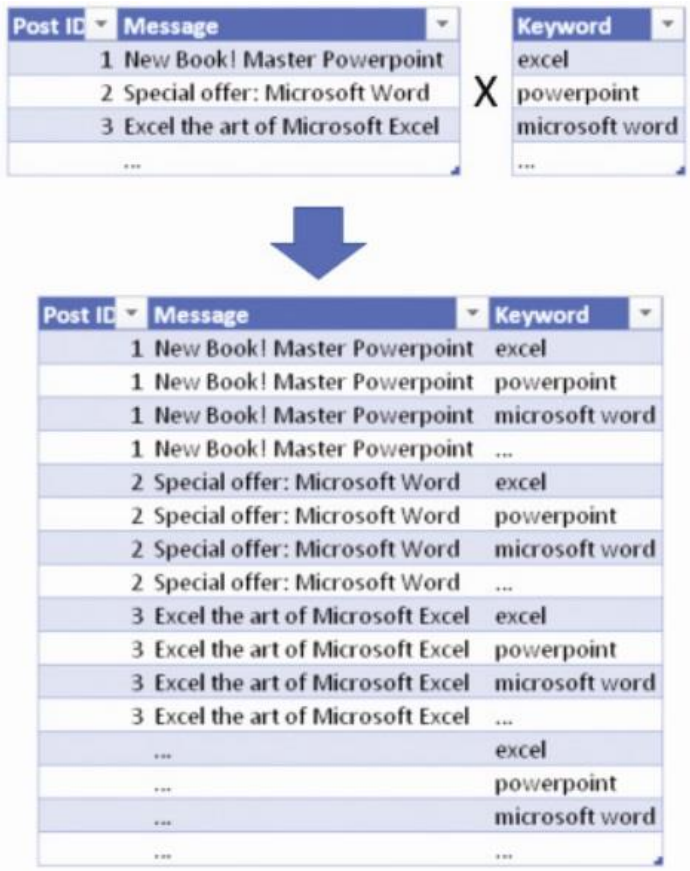


Рис. 3. Применение декартова произведения к сообщениям и ключевым словам

Загрузите файл C11E01.xlsx и сохранить его в папке C:\Data\C11\. Откройте новую книгу Excel. Пройдите *Данные –> Получить данные –> Из файла –> Из книги*. Выберите файл C11E01.xslx и щелкните *Импорт*. В окне *Навигатор* выберите *Sheet1* и щелкните *Преобразовать данные*. Переименуйте запрос в *Microsoft Press Posts*.

Создайте копию запроса *Microsoft Press Posts* для формирования новой таблицы сообщений и тем, причем один и тот же идентификатор поста (Post ID) может включать несколько строк. На панели *Запросы* щелкните правой кнопкой мыши на запросе *Microsoft Press Posts* и выберите *Ссылка*. Переименуйте новый запрос в *Post Topics*. Пройдите *Главная –> Выбор столбцов*. Оставьте *Post ID* и *Message,* щелкните Ok.

Выделите столбец *Message*, пройдите *Добавление столбца –> Создать дубликат столбца*. Выделите столбец *Копия Message*, пройдите *Преобразование –> Формат –> нижний регистр*. Переименуйте столбец *Копия Message* в *нижний регистр*.

Загрузите текстовый файл Keywords.txt и сохраните его в папке C:\Data\C11\. Для загрузки ключевых слов как нового запроса, при открытом редакторе Power Query пройдите *Главная –> Новый источник –> Создать запрос –> Файл –> Текстовый или CSV-файл.* Выберите файл Keywords.txt, щелкните *Импорт*. Откроется окно Keywords.txt. Щелкните Ok. В запросе *Keywords* переименуйте столбец *Column1* в *Keyword* . Выберите столбец *Keyword* и пройдите *Преобразование –> Формат –> нижний регистр*.

На панели *Запросы* выберите *Post Topics*. Пройдите *Добавление столбца –> Настраиваемый столбец*. В окне *Настраиваемый столбец* в поле *Имя нового столбца* введите *Cartesian*. В окне *Пользовательская формула столбца* введите:

= Keywords

Щелкните Ok. В таблице *Post Topics* появится столбец *Cartesian*, с таблицей *Keywords* в каждой строке. Щелкните на значке *Развернуть* в заголовке столбца *Cartesian* (или выделите столбец *Cartesian* и пройдите *Преобразование –> Развернуть*). Отмените установку флажка *Использовать имя исходного столбца в качестве префикса*. Щелкните Ok. Переименуйте столбец *Column1* в *Keyword*.

Обратите внимание – декартово произведение успешно реализовано. Для каждого Post ID имеется столько строк, сколько ключевых слов. Пройдите *Добавление столбца –> Условный столбец*. Настройте окно *Добавление условного столбца*:

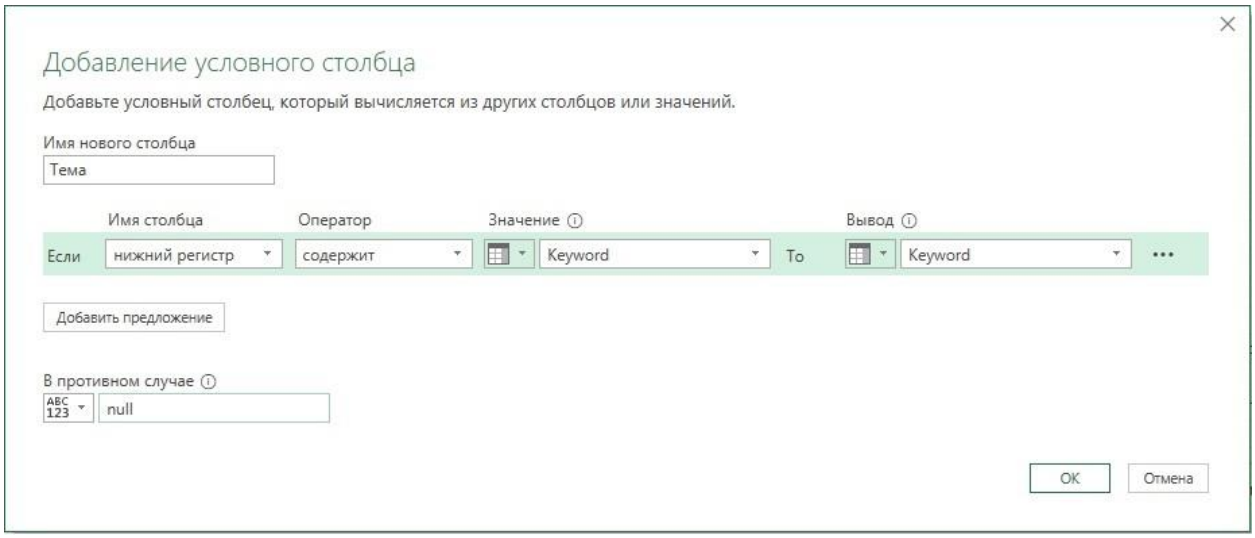


Рис. 4. Окно *Добавление условного столбца*

Щелкните Ok. В таблице появится столбец *Тема*. Щелкните на значке фильтра в столбце *Тема* и на панели *Фильтр* выберите команду *Удалить пустые*. Удалите столбцы *Message*, *нижний регистр* и *Keyword*. Переименуйте столбец *Тема* в *Keyword*. Используя декартово произведение, вы смогли сопоставить сообщения с определенными темами из динамического списка. Для создания отчета понадобятся две таблицы: *Microsoft Press Posts* (таблица фактов) и *Post Topics* (таблица измерений/подстановки).

Теперь можно перейти от подготовки данных к моделированию в Power Pivot (подробнее см. Роб Колли. Формулы DAX для Power Pivot и Альберто Феррари, Марко Руссо. Анализ данных при помощи Microsoft Power BI и Power Pivot для Excel). В редакторе PQ пройдите *Главная –> Закрыть и загрузить в…* В окне *Импорт данных* выберите две опции: *Только создать подключение* и *Добавить эти данные в модель данных*. Если до этого вы не загружали запросы в Excel, то теперь с такими опциями будут загружены все три запроса: *Microsoft Press Posts*, *Post Topics* и *Keywords*.

Если ранее загружали запросы в Excel, запросы загрузятся с ранее выбранными установками. Не беда. В Excel на панели *Запросы и подключения* кликните правой кнопкой последовательно на запросах *Microsoft Press Posts* и *Post Topics,* выберите *Загрузить в…* Настройте параметры загрузки: *Только создать подключение* и *Добавить эти данные в модель данных*. Для запроса *Keywords* выберите *Только создать подключение*. Запрос *Keywords* не потребуется в нашей модели данных.

В Excel пройдите *Power Pivot –> Управление*. В окне Power Pivot пройдите *Главная –> Представление диаграммы*. Для установления связи между двумя таблицами перетащите поле *Post ID* из таблицы *Microsoft Press Posts* на поле *Post ID* в таблице *Post Topics*.

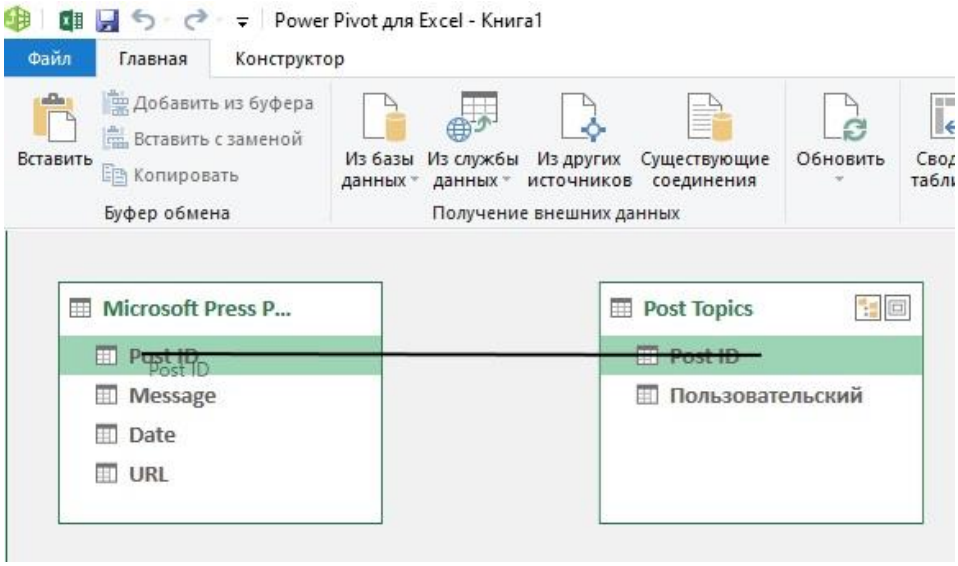


Рис. 5. Установите связь между двумя таблицами в Power Pivot

Теперь вы можете создать сводную таблицу и сводную диаграмму, основанную на обеих таблицах: *Microsoft Press Posts*, *Post Topics.* Например, такую:

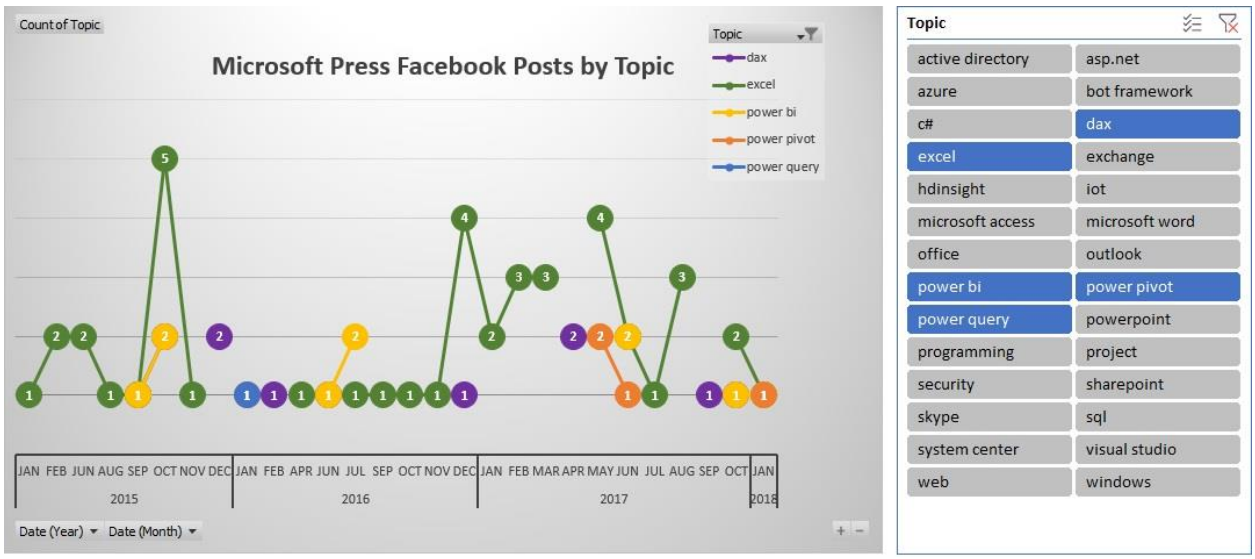


Рис. 6. Анализ постов Microsoft Press Facebook по темам (см. файл C11E02 - Solution.xslx)

Повышение производительности

Вы можете улучшить производительность декартова произведения. Откройте файл, с которым вы работали в предыдущем упражнении. В окне редактора Power Query выберите запрос *Post Topics* и шаг *Добавлен пользовательский объект*. В строке формул увидите:

= Table.AddColumn(#"Переименованные столбцы", "Cartesian", each Keywords)

Механизм языка M обращается к ключевым словам для каждой строки. Удивительно, но способ, которым язык M реализует этот шаг вместе с последующим шагом раскрытия, приводит к избыточным загрузкам ключевых слов из внешнего текстового файла. Чтобы внешний текстовый файл загружался только один раз можно использовать для ключевых слов M-функцию *Table.Buffer*. Тем самым вы явно указываете на необходимость сохранять таблицу ключевых слов в памяти и гарантируете, что механизм M не будет обращаться к текстовому файлу несколько раз.

> Несмотря на то, что функция Table.Buffer удобна для сокращения времени загрузки в этом упражнении, не следует применять ее всякий раз, когда необходимо объединить две таблицы. Например, при объединении таблицы с помощью команды *Объединить запросы* или *Добавить запросы* функция Table.Buffer не оказывает существенной помощи. Однако эта функция может помочь при использовании настраиваемых столбцов, для вычислений которых требуется доступ ко второму источнику данных.

Пройдите *Главная –> Расширенный редактор* и добавьте следующую строку после строки let:

KeywordsBuffer = Table.Buffer(Keywords),

Затем замените строку:

#"Добавлен пользовательский объект" = Table.AddColumn(
#"Переименованные столбцы", "Cartesian", each Keywords),

… на

#"Добавлен пользовательский объект" = Table.AddColumn(
#"Переименованные столбцы", "Cartesian", each KeywordsBuffer),

На этих небольших данных вы не заметите ускорения. Но вы можете загрузить файл C11E02 - Refresh Comparison.xlsx, который содержит существенно больше данных. На нем ускорение, связанное с использованием функции Table.Buffer, будет заметно.

Для оценки уменьшения времени обновления рекомендуется протестировать запрос на достаточно больших наборах данных. Для того чтобы искусственно увеличить имеющийся набор данных, можно умножить элементы таблицы с помощью функции Table.Repeat. В книге C11E02 - Refresh Comparison.xlsx можно умножить запрос Microsoft Press Posts 100 раз, применяя следующую функцию в качестве последнего шага запроса:

= Table.Repeat(#"Changed Type", 100).

Определение ключевых слов с помощью настраиваемой функции

Метод декартовых произведений не интуитивен, поэтому предложу еще один вариант – пользовательскую функцию. Она более понятна, но работает медленнее. Впрочем, если применить функцию Table.Buffer, то быстродействие этого метода повысится.

Откройте файл C11E02 - Refresh Comparison.xlsx и запустите редактор Power Query. На панели *Запросы* щелкните правой кнопкой мыши на *Post Topics* и выберите *Дублировать*. Переименуйте новый запрос в *Post Topics with Function*. Удалите последние пять шагов этого запроса. Пройдите *Главная –> Создать источник –> Другие источники –> Пустой запрос*. Переименуйте новый запрос в *FnDetectKeywords* и пройдите команду *Главная –> Расширенный редактор*. Удалите содержимое окна и вставьте код:

(Message, Keywords)=> Table.SelectRows(Keywords, each Text.Contains(Message, [Keyword]))

Эта функция получает сообщение и таблицу ключевых слов на вход возвращает подмножество таблицы *Keywords* с ключевыми словами, содержащимися в сообщении. Обратите внимание, предполагается, что таблица *Keywords* содержит столбец *Keyword*. Щелкните *Готово*.

На панели *Запросы* выберите *Post Topics with Functions*. Пройдите *Добавление столбца –> Настраиваемый столбец*. Введите *Keywords* в поле *Имя нового столбца* и следующую формулу в поле *Пользовательская формула столбца*:

= FnDetectKeywords([Lowercased], Keywords)

Щелкните Ok. Новый настраиваемый столбец определяет ключевые слова в каждой ячейке столбца Lowercased. Разверните столбец *Keywords*. (Снимите галочку *Использовать исходное имя столбца как префикс*.) Новый столбец *Keyword* содержит обнаруженные темы. В строках без соответствующих ключевых слов отображаются значения null. Щелкните на элементе управления фильтром в заголовке столбца *Keyword* и выберите команду *Удалить пустые.* Удалите столбцы *Message* и *Lowercased*. Загрузите запрос в таблицу на лист Excel.

Можно последовательно обновить запросы: *Post Topics – Faster*, *Post Topics* и *Post Topics with Function*. Запросы перечислены от самого быстрого к самому медленному. Исследуем, каким образом функция Table.Buffer улучшает производительность функции *FnDetectKeywords*. Откройте окно редактора Power Query и на панели *Запросы* правой кнопкой мыши щелкните на запросе *Post Topics with Function*. Выберите *Дублировать* и переименуйте новый запрос в *Post Topics - Fastest*. Пройдите *Главная –> Расширенный редактор*. Добавьте следующую строку после let:

KeywordsBuffer = Table.Buffer(Keywords),

Замените строку…

```
#"Добавлен пользовательский объект" = Table.AddColumn(
#"Renamed Columns", "Keywords", each FnDetectKeywords(
[Lowercased], Keywords)),
```

… на

```
#"Добавлен пользовательский объект" = Table.AddColumn(
#"Renamed Columns", "Keywords", each FnDetectKeywords(
[Lowercased], KeywordsBuffer)),
```

Загрузите запрос в таблицу на лист Excel и сравните время, необходимое для обновления четырех запросов. Запрос *Post Topics – Fastest* самый быстрый. Файл решения C11E03 - Solution.xlsx.

Разбиение на слова

Вспомните четвертую проблему, связанную со статическим поиском ключевых слов: слова, содержащиеся в качестве подстрок более длинных слов, могут ошибочно попадать в результат поиска. Чтобы улучшить обнаружение ключевых слов и получить новое представление о текстовых столбцах рассмотрим, как разбить текстовые столбцы на слова и отобразить результаты в сводных таблицах и диаграммах.

Начнем с интуитивно понятного подхода разделения на слова с помощью пробелов. Откройте новую книгу Excel. Пройдите *Данные –> Получить данные –> Из файла –> Из книги*. Выберите файл C11E01.xslx и щелкните *Импорт*. В окне *Навигатор* выберите *Sheet1* и щелкните *Преобразовать данные*. Переименуйте запрос в *Microsoft Press Posts*. Создадим новую таблицу из ID постов и слов. На панели *Запросы* щелкните правой кнопкой мыши на запросе *Microsoft Press Posts* и выберите *Ссылка*. Переименуйте новый запрос в *All Words*.

Выделите запрос *All Words* и пройдите *Главная –> Выбор столбцов*. Оставьте столбцы *Post ID* и *Message*. Щелкните Ok. Правой кнопкой мыши щелкните на столбце *Message* и выберите *Разделить столбец –> По разделителю*. Как было показано ранее, разбиение текстового столбца на столбцы (по умолчанию) плохо масштабируется, тем более что в столбце *Message* очень много слов. Разделить столбец *Message* на строки:

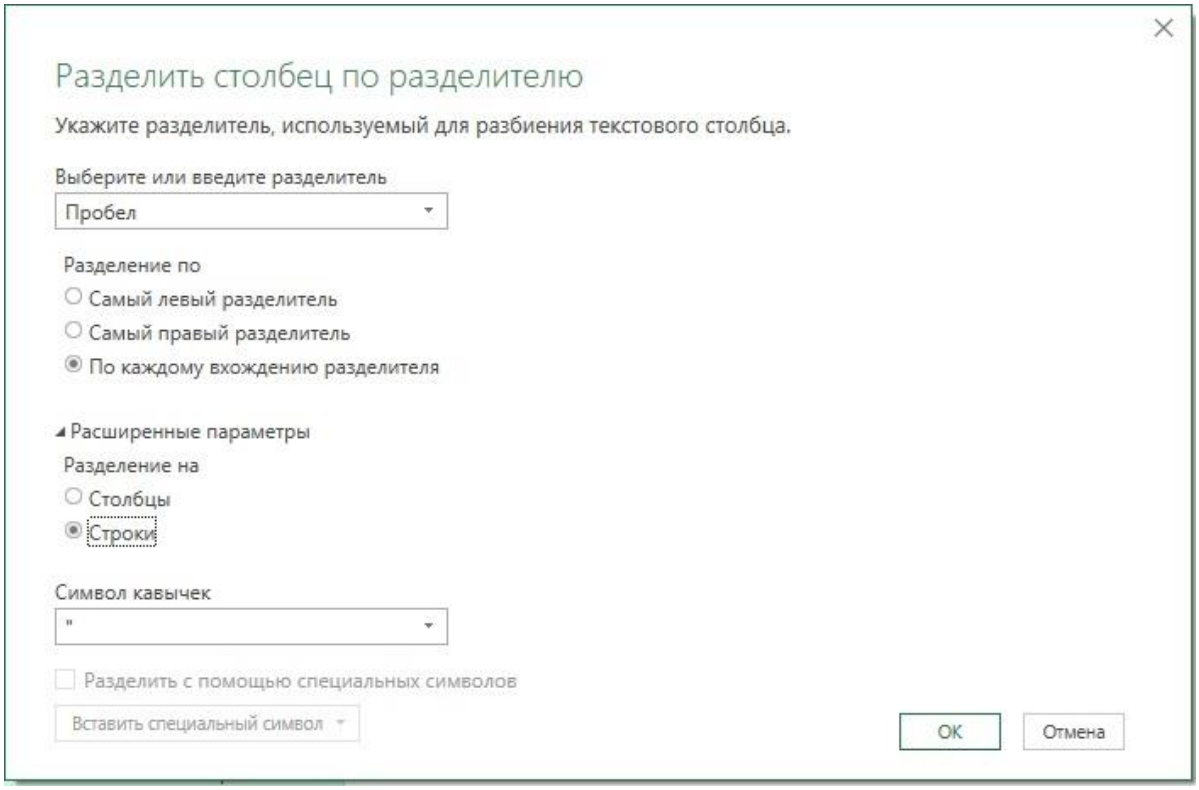


Рис. 7. Окно *Разделить столбец по разделителю*

Щелкните Ok. Переименуйте новый столбец в *Word*. К сожалению, многие слова содержат знаки препинания. Последние необходимо отделить.

Создайте пустой запрос, назовите его *Punctuations* и в строку формул редактора Power Query введите:

= {" ", "~", ",", ".", "?", "!", "(", ")", "[", "]", "{", "}", "@",
"#", "$", "%", "^", "&", "*", "-", "+", "=", ":", ";", "|",
"<", ">", "/", "'", "'", "\", "'", "#(tab)", "#(cr)", "#(lf)"}

Это выражение создает список знаков препинания, а также специальных символов для обнаружения табуляции, возвратов каретки и конца строк.

На панели *Запросы* выберите *All Words*. Выберите шаг *Разделить столбец по разделителю*. В строке формул вы обнаружите довольно сложную формулу:

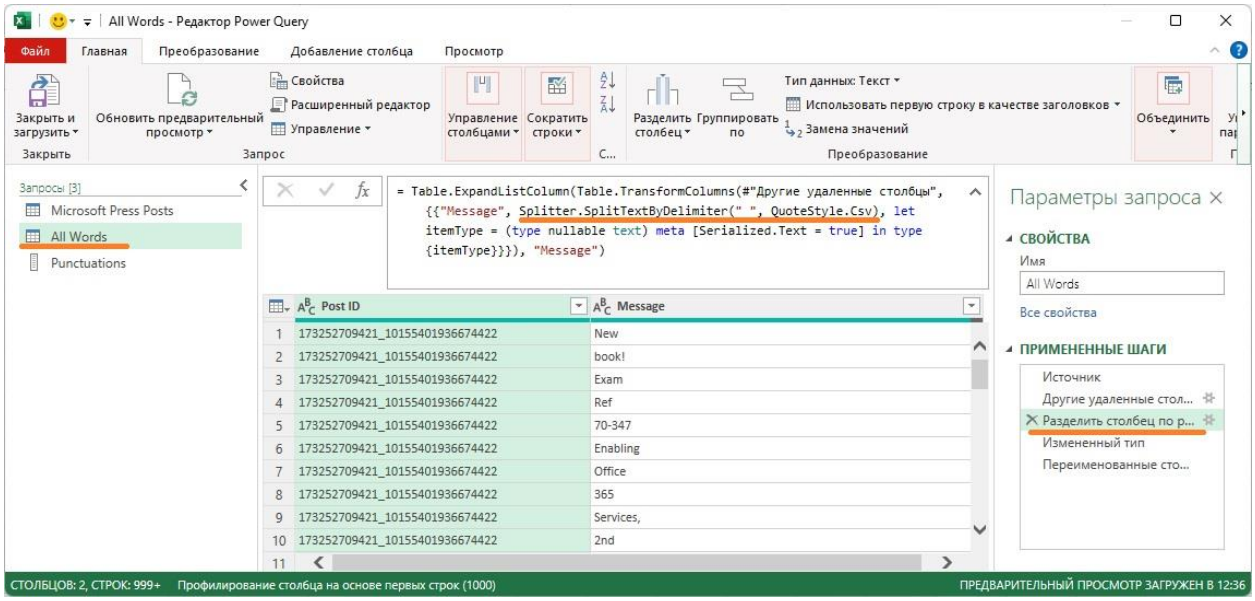


Рис. 8. Формула шага *Разделить столбец по разделителю*

Обратите внимание на выделенный фрагмент кода:

Splitter.SplitTextByDelimiter(" ", QuoteStyle.Csv)

Он определяет тип разделителя – пробел и стиль кавычек. Измените код на…

Splitter.SplitTextByAnyDelimiter(Punctuations, QuoteStyle.Csv)

Эта модификация выполняет разбиение по всем разделителям из таблицы *Punctuations*.

> Если необходимо разбить столбец с учетом нескольких разделителей, можно использовать команду *Разделить столбец по разделителю* в интерфейсе пользователя для разделения с помощью одного разделителя, а затем модифицировать формулу. Замените функцию Splitter.SplitTextByDelimiter на Splitter.SplitTextByAnyDelimiter и задайте список текстовых значений вместо одного разделителя. Этот подход пригоден при разбиении столбца на столбцы или строки.

Из-за разделения с помощью знаков препинания появилось большое число пустых строк. Выполним их фильтрацию. Оставаясь в запросе *All Words* выберите шаг Переименованные столбцы и щелкните мышью на элементе управления фильтром в заголовке столбца *Word*. На панели *Фильтр* выберите команду *Удалить пустые*. Загрузите в модель данных запросы *Microsoft Press Posts* и *All Words*. Для запроса *Punctuations* установите *Только создать подключение* (он не нужен нам в модели данных). В Power Pivot создайте связь между полями *Post ID* в двух таблицах. Закройте Power Pivot. В Excel создайте сводную таблицу для отражения частоты слов:

	A	B
1	Слова	Число элементов
2	http	603
3	The	475
4	and	465
5	msft	432
6	to	426
7	Microsoft	379
8	on	313
9	it	288
10	of	269
11	for	266

Рис. 9. Частота слов в сообщениях *Microsoft Press*

Видно, что в качестве отдельных слов используется много служебных. Сохраните файл. Он понадобится нам позже.

Разделение слов пробелами и удаление пунктуации

Усовершенствуем метод. Вместо разбивки сообщений с помощью любого из знаков препинания, будем разбивать сообщения только при появлении пробела, знака табуляции, возврата каретки и перевода строки. Затем, если знаки препинания обнаружатся в начале или конце слов, вы их обрежете.

Откройте файл, сохраненный в предыдущем упражнении. Запустите редактор Power Query. На панели *Запросы* щелкните правой кнопкой мыши на *All Words* и выберите *Дублировать*. Переименуйте новый запрос в *Trim Punctuations*. В этом запросе выберите шаг *Разделить столбец по разделителю*. В строке формул вместо *Punctuations* введите список разделителей. Строка примет вид:

```
= Table.ExpandListColumn(Table.TransformColumns(
#"Другие удаленные столбцы", {
{"Message", Splitter.SplitTextByAnyDelimiter(
{" ", "#(tab)", "#(cr)", "#(lf)"}, QuoteStyle.Csv),
let itemType = (type nullable text) meta
[Serialized.Text = true] in type {itemType}}}), "Message")
```

Оставаясь на шаге *Разделить столбец по разделителю* выделите столбец *Message,* пройдите *Преобразование –> Формат –> Усечь*. Этот шаг по умолчанию удаляет пробелы в начале и конце текста. Модифицируем формулу, чтобы обрезать все знаки препинания.

Замените формулу:

```
= Table.TransformColumns(#"Разделить столбец по разделителю",
{{"Message", Text.Trim, type text}})
```

… на

```
= Table.TransformColumns(#"Разделить столбец по разделителю",
{{"Message", each Text.Trim(_, Punctuations), type text}})
```
[1]

Вместо заданной по умолчанию функции Text.Trim можно явно определить новую одноименную функцию Text.Trim с другими аргументами. Необходимую функцию формирует ключевое слово each. Вторым аргументом функции Text.Trim определяется список символов или строк для обрезки. В данном случае в качестве аргумента применяется список Punctuations.

> Для повышения эффективности, поскольку на шаге разделения уже использованы пробел, символ табуляции, символ возврата каретки и символ новой строки, вы можете

[1] Почему-то этот код у меня не работал, хотя авторская версия в файле C11E04 - Solution.xlsx полностью аналогична и работоспособна.

сформировать дублирующий список пунктуации и удалить эти символы из нового списка. Затем можно применить новый список в качестве второго аргумента функции Text.Trim.

Теперь при создании сводной таблицы http более не встречается. Сохраните отчет. Он понадобится в следующем упражнении. Файл решения C11E04 - Solution.xlsx.

Фильтрация по стоп-словам

В предыдущем упражнении было показано, как разбивать сообщения на слова, но в итоге получен тривиальный результат. Наиболее часто встречающиеся слова — это союзы, предлоги и артикли (см. рис. 9). Чтобы отфильтровать эти стоп-слова можно сформировать их список в отдельном файле, загрузить их в запрос и выполнить фильтрацию.

Откройте финальный файл с прошлого упражнения или книгу C11E04 - Solution.xlsx (в дальнейшем изложении я использую именно этот файл). Запустите редактор Power Query, на панели *Запросы* правой кнопкой мыши щелкните на *All Words - Trim Punctuations*. Выберите *Ссылка*. Назовите новый запрос *No Stop Words*. Загрузите файл Stop Words.txt из приложенного архива и сохраните его в папке C:\Data\C11\. Импортируйте файл Stop Words.txt, находясь в редакторе PQ и пройдя *Главная –> Создать источник –> Файл –> Текстовый или CSV файл*.

Убедитесь, что новый запрос называется *Stop Words*, и переименуйте столбец *Column1* в *Stop Word*. Выберите столбец *Stop Word* и пройдите *Преобразование –> Формат –> нижний регистр*. Всегда работайте с нижним регистром, чтобы избегать неоднозначностей.

Перейдите в запрос *No Stop Words*. Столбец *Word* содержит слова в нижнем регистре. Объединим слова и стоп-слова. Пройдите *Главная –> Объединить запросы*. Настройте окно *Слияние*

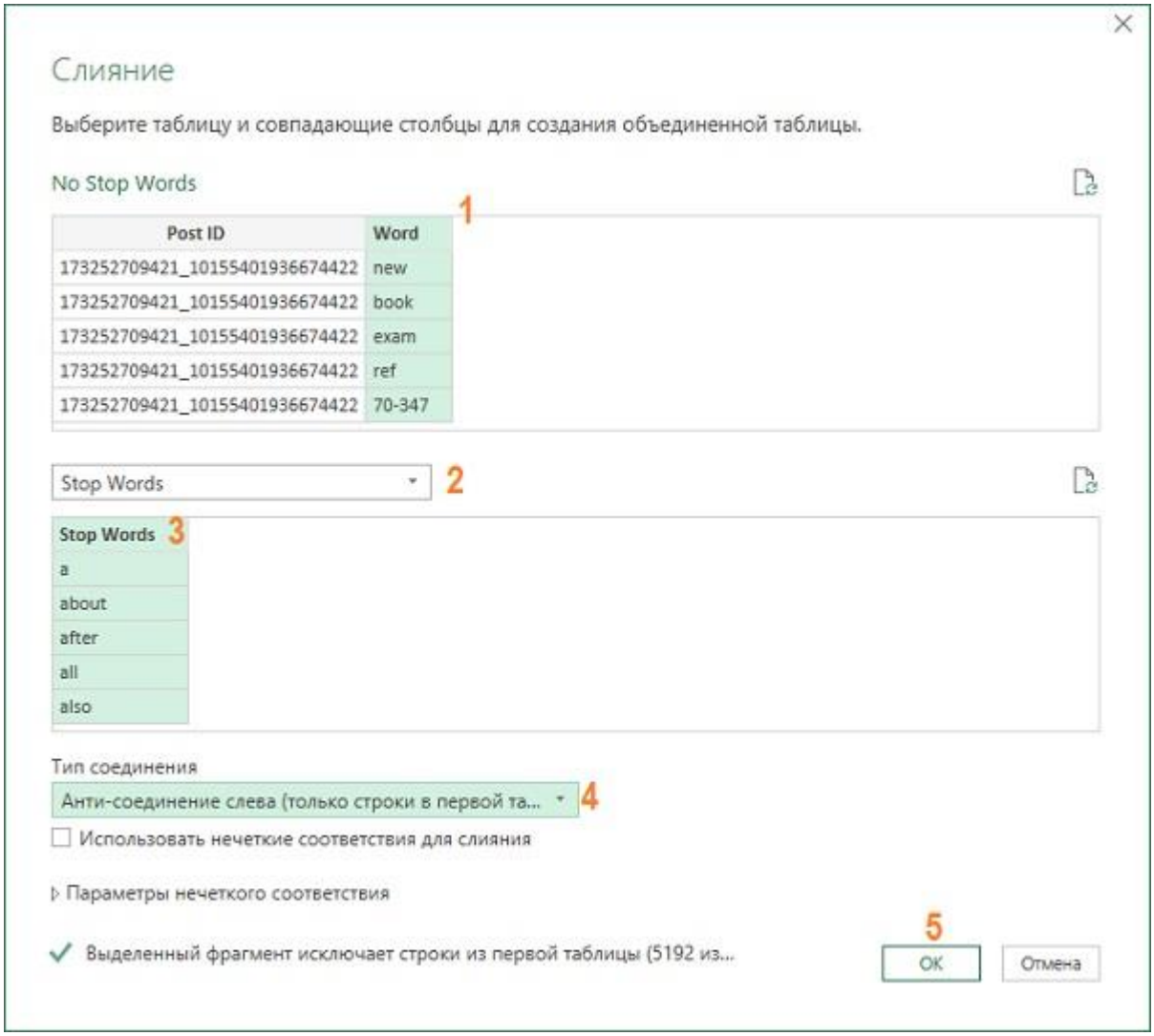


Рис. 10. Окно объединения двух таблиц

При слиянии столбец *Stop Words* объединяется с объектами таблицы. Если щелкнуть на пустую область любой ячейки столбца *Stop Words* (но не на ссылку таблицы), то внизу увидите, что все таблицы содержат нулевые значения. Нулевые значения отображаются вследствие выполнения операции левого антислияния, которая не возвращает строк из второй таблицы слияния. Из запроса *Stop Words* нечего извлекать.

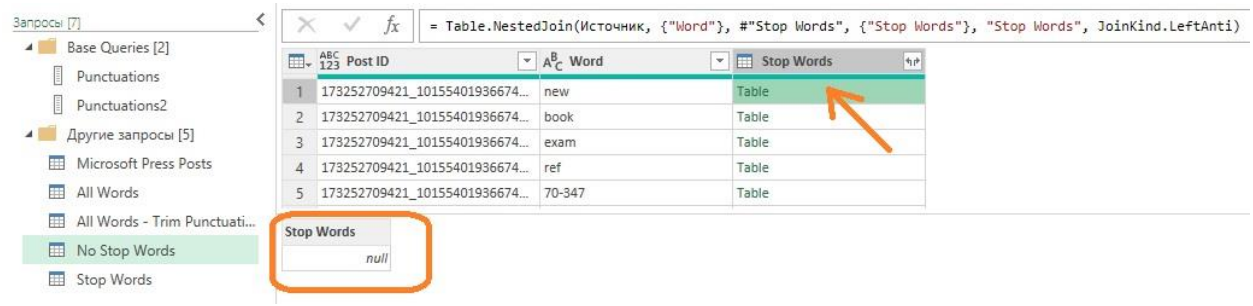


Рис. 11. Результат объединения двух таблиц

Удалите столбец *Stop Words* и загрузите запросы: Stop Words – *Только создать подключение*, а *No Stop Words* – *Только создать подключение* и *Добавить эти данные в модель данных*.

В Excel создайте сводную таблицу. Отсортируйте слова по популярности по убыванию и отразите только ТОП-10:

Слово	Число вхождений
microsoft	370
new	248
windows	238
save	221
50	169
exam	157
book	155
2016	147
saturday	134
ebook	128
Общий итог	**1967**

Рис. 12. Десятка самых популярных слов в постах Microsoft Press без стоп-слов

Одним из наиболее распространенных слов является *2016*, которое отражает большое количество сообщений Microsoft о программных продуктах 2016 (Office 2016, Excel 2016, Windows Server 2016). Другое распространенное слово – *50*. Оно столь популярно из-за рекламы книги со скидкой 50%. Знаете ли вы, что по субботам вы можете получить 50% скидки на книги Microsoft Press?

Сохраните файл для использования в следующем упражнении. Файл решения C11E05 - Solution.xlsx.

Поиск по ключевым словам с помощью разделенных слов

В предыдущих упражнениях ключевые слова разыскивались в постах Microsoft Press с помощью оператора *contains* языка M. Один из недостатков этого подхода состоит в том, что можно прийти к ложному обнаружению слов, если ключевое слово является подстрокой другого слова. Например, ключевое слово *excel* может быть подстрокой слова *excellent*. Изучим более точный метод. Начнем с рассмотрения полезной техники для фильтрации таблицы по нескольким значениям путем слияния запросов.

Откройте финальный файл с прошлого упражнения или книгу C11E05 - Solution.xlsx (далее я использую этот файл). Запустите редактор Power Query и пройдя *Главная –> Создать источник –> Файл –> Текстовый или CSV файл*. Импортируйте файл *Keywords.txt*. Переименуйте столбец *Column1* в *Keyword*. Чтобы все ключевые слова находились в нижнем регистре, выделите столбец *Keyword* и пройдите *Преобразование –> Формат –> нижний регистр*.

На панели *Запросы* щелкните правой кнопкой мыши на *No Stop Words* и выберите *Ссылка*. Переименуйте запрос в *Post Topics*. Пройдите *Главная –> Объединить запросы*. Настройте окно Слияние:

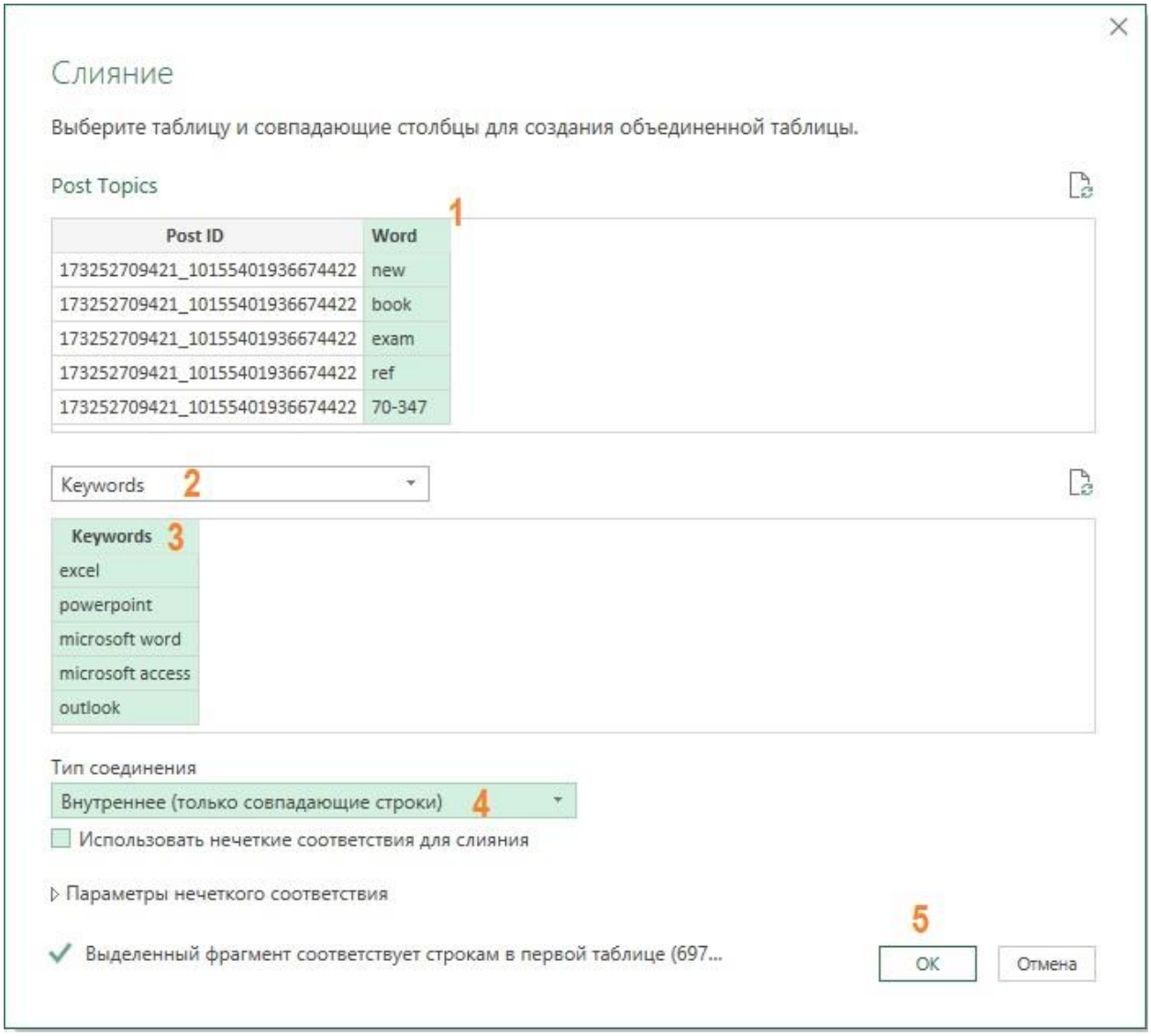


Рис. 13. Параметры настройки окна *Слияние*

Переименуйте столбец *Word* в *Topic*. Чтобы избежать ситуаций, когда одно ключевое слово несколько раз упоминается в одном сообщении, можно выделить столбцы *Post ID* и *Topic* и пройти *Главная –> Удалить строки –> Удалить дубликаты*. Теперь каждая тема упоминается только один раз для каждого *Post ID*.

На этом этапе можно и завершить упражнение. Но появилась новая проблема: заметили ли вы, что некоторые ключевые слова в файле *Keywords.txt* состоят из двух слов, например power query, power bi и power pivot? Поскольку это упражнение начато с разделения постов на отдельные слова, вы не сможете обнаружить ключевые слова, состоящие из двух слов. В следующем разделе показано, как разрешить это затруднение.

Сохраните файл для последующего использования.

Обнаружение составных ключевых слов

Для определения таких ключевых слов, как power query и power bi в постах Microsoft Press в этом упражнении, нужно временно заменить пробел другим символом, например подчеркиванием, но только при обнаружении ключевых слов, состоящих из двух слов в столбце Message. Затем можно применить разделение на пробелы, которые больше не будут влиять на ключевые слова из двух слов. Как только сообщения разбиты на слова, можно преобразовать ключевые слова, состоящие из двух слов, обратно в форму, разделенную пробелами.

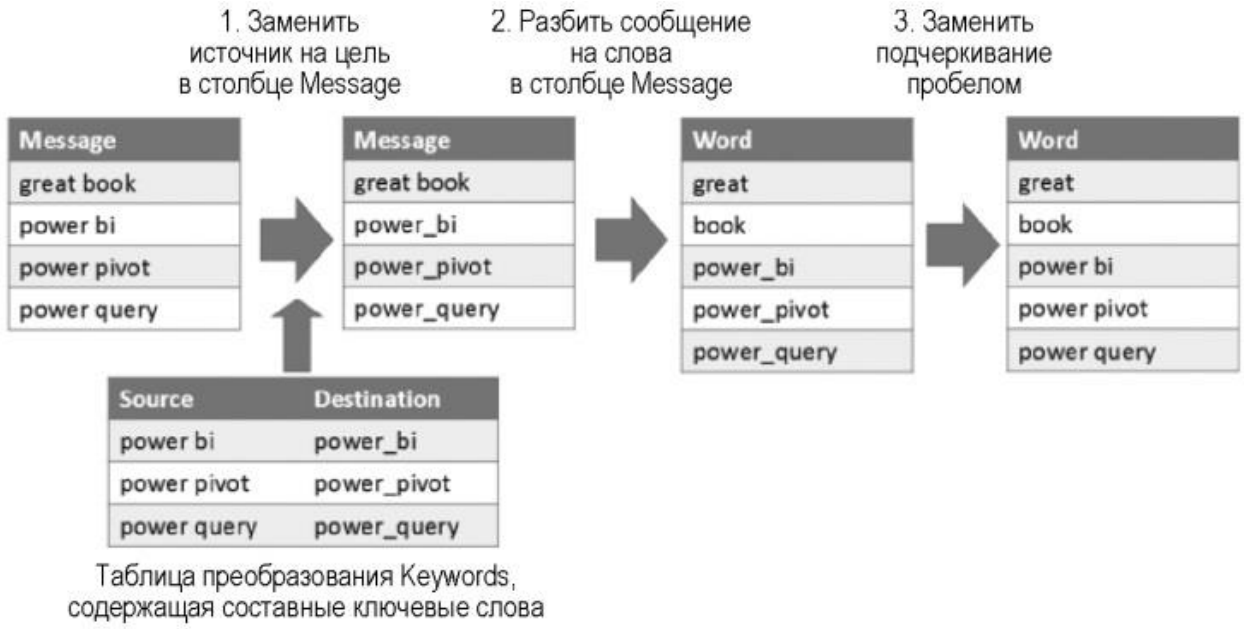


Рис. 14. Шаги метода

Откройте файл с предыдущего упражнения. Перейдите в редактор Power Query. На панели *Запросы* кликните правой кнопкой мыши на *Keywords* и выберите *Ссылка*. Переименуйте запрос в *Conversion Table*. Выделите столбец *Keywords*. Пройдите *Преобразование –> Формат –> Усечь*. Даже если файл *Keywords* не содержит начальных или конечных пробелов, рекомендуется выполнить эту команду перед выполнением следующего шага. Щелкните мышью на элементе управления фильтром в заголовке столбца *Keyword*. На панели *Фильтр* выберите *Текстовые фильтры –> Содержит –> пробел*. Щелкните Ok. На панели предварительного просмотра отобразятся девять строк с ключевыми словами, состоящими из двух слов.

Переименуйте столбец *Keyword* в *Source*. Выделите его. Пройдите *Добавление столбца –> Создать дубликат столбца*. Переименуйте новый столбец в *Destination*. Выделите его. Пройдите *Главная –> Замена значений*. В окне *Замена значений* выберите символ пробела в поле *Значение для поиска* и символ подчеркивания в поле *Заменить на*. Щелкните Ok.

Таблица преобразования *Keyword*, содержащая составные слова (см. рис. 14) подготовлена. Далее необходимо заменить каждое значение *Source* на соответствующее значение *Destination* для каждого значения столбца *Message* перед выполнением шага *Split*. Замену одного значения легко выполнить в интерфейсе редактора Power Query, но в данном случае имеется несколько пар *Source/Destination*. К сожалению, в редакторе Power Query отсутствует кнопка пользовательского интерфейса для применения нескольких замен строк. Кроме того, функция M Table.ReplaceValue не имеет необходимого нам свойства. Можно воспользоваться усовершенствованной методикой, основанной на применении функции List.Accumulate для выполнения нескольких замен текста из *Source* на значения *Destination*.

Функция List.Accumulate рассматривалась ранее. Она получает список в качестве входных данных, начальный объект и функцию аккумулятора. Функция аккумулятора должна иметь объект состояния и текущий объект (который является членом списка ввода на текущей итерации). Эта функция возвращает объект состояния после его преобразования. В заключение функция List.Accumulate возвращает накопленный результат после того, как функция аккумулятора перебрала список.

Прежде чем реализовать функцию List.Accumulate, рассмотрим, каким образом можно преобразовать запрос таблицы преобразования в список пар *Source/Destination*, которые можно применять в качестве входных данных для функции List.Accumulate. Ниже показано, как сформировать список с помощью функции Table.ToRecords, которая разработана для возврата списка записей (одна запись на одной строке исходной таблицы).

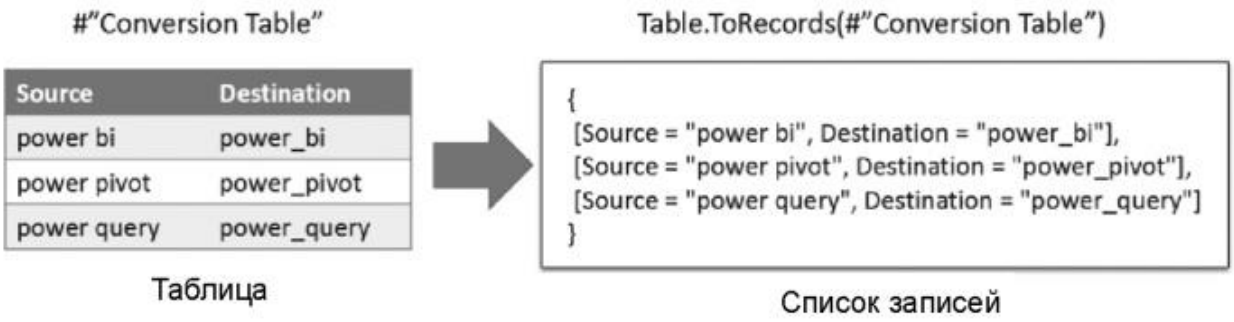


Рис. 15. Функция M Table.ToRecords может трансформировать таблицу преобразования в список записей

Функция List.Accumulate получает сообщение Facebook Post и заменяет текущий текст из столбца *Source* соответствующим текстом из столбца *Destination*. Например, power bi заменяется на power_bi, что позволит избежать в дальнейшем разбиения ключевого слова power bi на два отдельных слова:

```
List.Accumulate(
        Table.ToRecords(#"Conversion Table"),
        [Message],
        (message, conversionRecord) =>
                Text.Replace(
                        message,
                        conversionRecord[Source],
                        conversionRecord[Destination]
                )
)
```

Если формула применяется в пользовательском столбце, то она выполняет перебор каждой строки и замену значения в столбце Message, выполняя просмотр всех ключевых слов в столбце *Source* и замену их одно за другим соответствующими ключевыми словами из столбца *Destination*.

Итак, в окне Power Query на панели *Запросы* выберите *All Words - Trim Punctuations*. Этот запрос выполняет разделение слова. Выберите шаг *Lowercased Text*. Пройдите *Добавление столбца –> Настраиваемый столбец*. Подтвердите вставку шага. Настройте параметры:

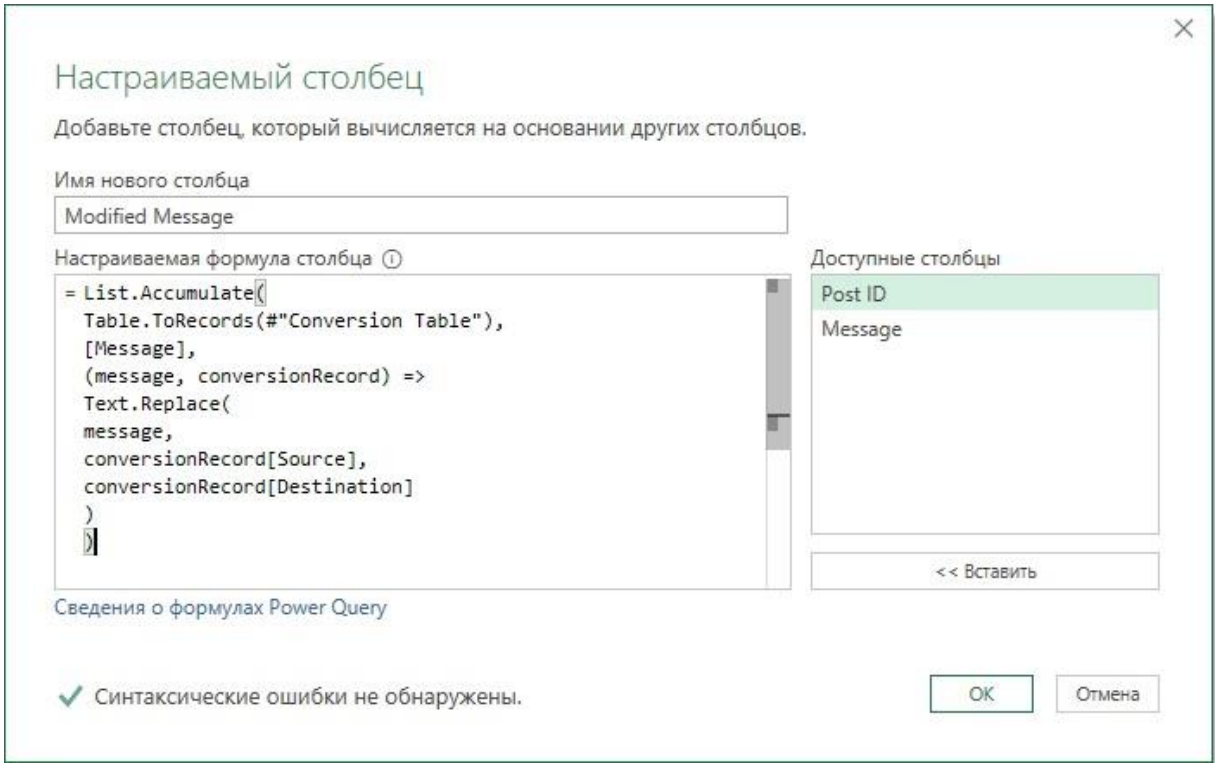


Рис. 16. Окно *Настраиваемый столбец*

Щелкните Ok. Столбец *Modified Message* включает посты Facebook с замененными значениями. Чтобы протестировать работу функции List.Accumulate добавьте временный шаг – отфильтруйте столбец *Message* по словам power query и power pivot. Столбец *Modified Message* бкдет содержать эти слова с подчеркиванием вместо пробела. Удалите шаг *Строки с примененным фильтром*.

Выберите шаг *Добавлен пользовательский столбец* и удалите столбец *Message*. Подтвердите вставку шага. Переименуйте столбец *Modified Message* в *Message*. Подтвердите вставку шага. Теперь все ключевые слова, состоящие из двух слов, содержат подчеркивание в качестве разделителя. Они защищены от разделения на следующем шаге преобразования, который уже присутствует на панели *Примененные шаги*.

Выберите запрос *No Stop Words* и перейдите к последнему шагу. Для удаления символа подчеркивания имеются две возможности. Первый вариант относительно прост: если можно предположить, что отсутствуют важные слова с подчеркиванием, можно применить команду *Замена значения* для столбца Word и заменить подчеркивания пробелами. Второй вариант более эффективен, если ваши данные содержат слова с подчеркиванием (не ключевые слова).

Пройдите *Настраиваемый столбец –> Добавление столбца*. Назовите новый столбец *Modified Word*. Вставьте код в поле *Пользовательская формула столбца*:

```
List.Accumulate(
        Table.ToRecords(#"Conversion Table"),
        [Word],
        (word, conversionRecord) =>
        if word = conversionRecord[Destination] then
                conversionRecord[Source]
        else
                word
)
```

Щелкните Ok.

Приведенная формула выполняет итерации по списку записей пар ключевых слов *Source/Destination*. Начальным состоянием этой функции (второй аргумент List.Accumulate) является значение в столбце *Word*. Функция аккумулятора (третий аргумент функции List.Accumulate) включает состояние и текущие аргументы (под названием word и conversionRecord). Аргумент состояния содержит значение в текущей ячейке столбца Word. Текущим аргументом функции аккумулятора является текущая запись пары *Source/Destination*. Функция аккумулятора сравнивает текущее слово и слово в столбце *Destination*. Если соответствие установлено, значение возвращается столбцу *Source*, потому что необходимо вернуть ключевое слово обратно из *Destination* в *Source*. Если же ключевое слово не соответствует *Destination*, то сохраняется исходное слово. На очередной итерации для функции List.Accumulate оценивается следующая пара *Source/Destination*.

Для тестирования формулы отфильтруйте столбец *Word* с помощью ключевого слова *power_bi*. Столбец *Modified Word* содержит power bi. После подтверждения, что предыдущий шаг выполнен правильно, удалите шаг фильтра. Удалите столбец *Word*. Переименуйте столбец *Modified Word* в *Word*. Выберите запрос *Post Topics* и щелкните на элементе управления фильтром столбца *Topic*. В панели *Фильтр* вы увидите, что найдены все ключевые слова, составленные из двух слов.

И наконец, перед загрузкой запроса *Post Topics* можно уменьшить время загрузки, применяя функции Table.Buffer или List.Buffer. Поскольку запросы используют настраиваемые столбцы, которые обращаются к таблице преобразования, механизм языка M может обращаться к внешним источникам несколько раз, при ссылках на внешние запросы внутри настраиваемого столбца.

Выберите запросе *All Words - Trim Punctuations*. Пройдите *Главная –> Расширенный редактор*. Перед строкой, которая начинается с #"Добавлен пользовательский объект", вставьте код

```
BufferedList = List.Buffer(Table.ToRecords(#"Conversion Table")),
```

На шаге #"Добавлен пользовательский объект" замените Table.ToRecords(#"Conversion Table") на BufferedList. Щелкните *Готово*.

Выберите запрос *No Stop Words*. Пройдите *Главная –> Расширенный редактор*. Перед строкой, которая начинается с #"Добавлен пользовательский объект", вставьте код

BufferedList = List.Buffer(Table.ToRecords(#"Conversion Table")),

На шаге #"Добавлен пользовательский объект" замените Table.ToRecords(#"Conversion Table") на BufferedList. Щелкните *Готово*.

Загрузите запрос *Post Topics* модель данных. Создайте сводную таблицу и сводную диаграмму для иллюстрации того, какие темы наиболее популярны на странице Microsoft Press Facebook. Файл решения C11E06 - Solution.xlsx.

Row Labels	Count of Topic
windows	202
azure	85
office	46
web	44
excel	41
sql	40
sharepoint	27
visual studio	21
programming	20
security	19
asp.net	17
active directory	16
microsoft word	15
exchange	14
project	14
system center	13
dax	12
power bi	12
outlook	8
bot framework	8
iot	8
powerpoint	7
microsoft access	7
power pivot	5
skype	2
power query	1
hdinsight	1
Grand Total	**705**

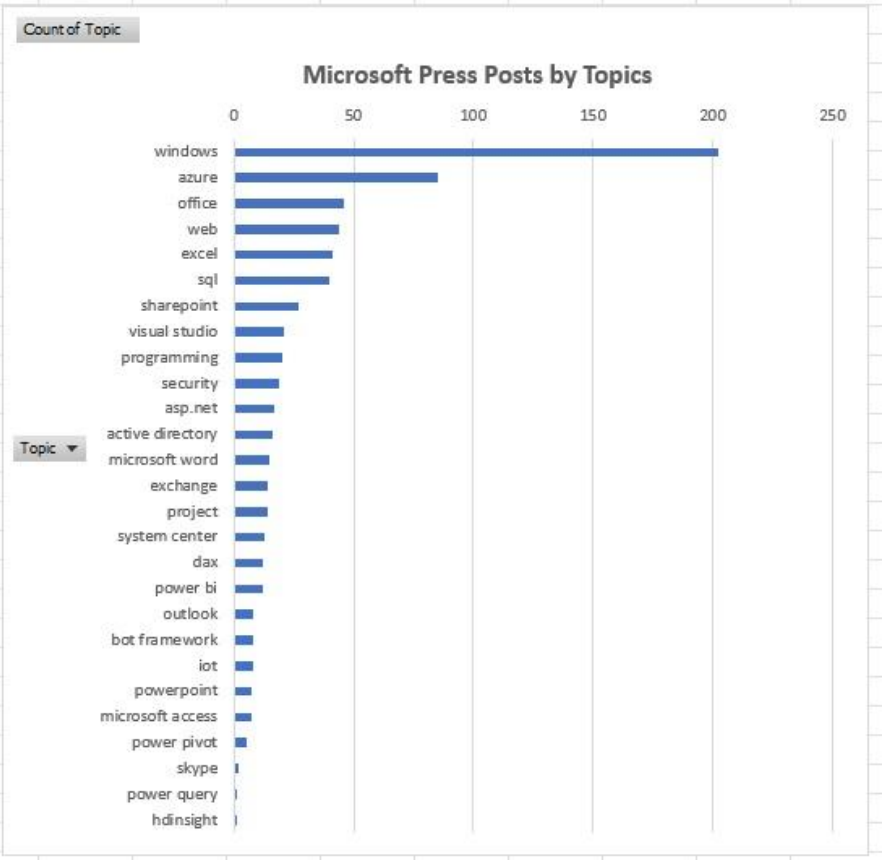


Рис. 17. Количество постов Facebook по темам в виде сводной таблицы и в сводной диаграммы