В.Н. Верещако (Минск, МГЛУ)

ПОДХОДЫ К АВТОМАТИЧЕСКОМУ ПОРОЖДЕНИЮ КОНТЕКСТНО-ЗАВИСИМЫХ АННОТАЦИЙ ИНТЕРНЕТ-ДОКУМЕНТОВ

С развитием сети Интернет появился новый вид аннотаций — контекстно-зависимые аннотации, которые являются очень актуаль-ными в контексте сетевого поиска информации [1]. В списке результатов работы поисковой машины наряду с заголовком и адресом найденного документа обычно присутствует сниппет (*snippet*) — фрагмент этого документа, содержащий слова запроса. Он позволяет пользователю принять решение о необходимости обращения ко всему первичному документу.

Формируемые поисковой системой контекстно-зависимые аннотации должны удовлетворять ряду требований. Их можно разделить на два вида: 1) требования, которые выдвигает пользователь, и 2) требования, которые накладываются самой поисковой системы. Среди требований, выдвигаемых пользователем, можно выделить следующие. Аннотация должна:

1) давать представление о том, какую информацию содержит найденный документ о предмете запроса;

2) быть достаточно короткой, чтобы ее анализ пользователем не занимал много времени;

3) давать представление о том, какие части документа (если он достаточно большой) несут релевантную информацию.

С точки зрения поисковой системы к алгоритму формирования аннотации предъявляются следующие требования:

1) конечная вычислительная трудоемкость. Создание аннотаций является, с точки зрения поисковой системы, вспомогательной задачей и не должна приводить к заметному замедлению ее работы;

2) максимальное использование информации, полученной при поиске на этапах отбора и «взвешивания» смысла документов;

3) приемлемый требуемый объем оперативной памяти.

В настоящее время известно много алгоритмов автоматического формирования контекстно-зависимых аннотаций. С учетом требования небольшой длины текста аннотации (практически не более 150–400 символов) в информационно-поисковых системах чаще всего используется следующий подход: аннотация содержит один непрерывный фрагмент текста документа. В данном случае основное внимание уделяется выбору оптимального фрагмента [2]. Алгоритм выбора наилучшего фрагмента исходного документа базируется на предположении, что найденный по запросу единый фрагмент связного текста отражает именно ту часть документа, которая необходима. Основным элементом для построения контекстно-зависимых аннотаций является фрагмент текста, удовлетворяющий следующим требованиям [3]:

1) он содержит наибольшее число слов запроса;

2) между двумя предложениями фрагмента, которые содержат слова запроса, может находиться не более одного предложения, не содержащего ни одного слова запроса.

Принципиальный алгоритм построения текста аннотации состоит из следующих этапов: 1) идентификации фрагментов найденного текста, удовлетворяющих вышеупомянутым условиям; 2) выбора фрагментов текста, содержащих наибольшее количество различных слов запроса (в случае существования фрагментов, содержащих равное число уникальных слов запроса, происходит оценка фрагмента на основе частоты вхождения слов фрагмента в исходный документ); 3) сравнения размера отобранного фрагмента исходного текста с максимально допустимым размером аннотации. В случае если размер фрагмента найденного текста меньше требуемого размера аннотации, последняя дополняется заголовком, предшествующим выбранному фрагменту текста, который помещается в ее начало. Если размер созданной аннотации все еще остается

меньше максимально допустимого, в нее включаются предложения, следующие в исходном документе после выбранного фрагмента, но не выходящие за границы параграфа, которому принадлежит выделенный фрагмент.

В рамках данного подхода выделяют три основных алгоритма формирования контекстно-зависимых аннотаций [2]:

1) базовый алгоритм, согласно которому в тексте найденного документа анализируются только слова запроса, и выбирается фрагмент с их наибольшей плотностью;

2) алгоритм под названием *Freq*, в котором учитываются не только слова запроса, но и наиболее частотные слова найденного текста, находящиеся рядом с анализируемым фрагментом;

3) алгоритм под названием *LRU-K*, в котором при выборе оптимального фрагмента учитываются, как слова запроса, так и слова, найденные по алгоритму определения повторяемости слов в тексте.

При использовании базового алгоритма происходит анализ текста документа, в ходе которого выбирается его самый «тяжелый» фрагмент фиксированной длины, не пересекающий границу абзаца. Вес фрагмента документа определяется по формуле $W_b = Sum(W_i) + K * n / L$, где $Sum(W_i)$ — сумма весов слов запроса, вошедших во фраг-мент. При этом каждое слово учитывается только один раз. Вес слова зависит от его распределения в тексте и является тем выше, чем более редко встречается это слово. Таким образом, приведенная выше формула оценивает каждый фрагмент найденного документа исходя из того, располагаются ли в нем слова запроса на минимальном расстоянии друг от друга, находится ли в нем слов запроса больше, чем в других фрагментах, причем выше оцениваются те слова запроса, которые реже встречаются во всем тексте. В список результатов поиска для найденного документа помещается тот фрагмент текста, который получает наибольший вес. Если несколько фрагментов получают одинаковые веса, то выводится тот, который находится ближе к началу текста. Отобранный фрагмент «выравнивается в тексте», то есть вырезается некоторое количество слов до первого слова запроса, и несколько после, формируя длину фрагмента до необходимой длины.

Описанный выше базовый алгоритм не может привести к формированию аннотаций высокого качества, если запрос содержит всего одно слово или часто встречающееся словосочетание. В этом случае он просто выводит первый фрагмент текста, в котором встретились слова запроса.

На основе анализа существующих методов составления контекстно-зависимых аннотаций для повышения эффективности работы базового алгоритма в работе [3] предлагаются следующие модификации.

1. Чаще всего количество слов запроса к поисковым системам не превышает трех. Если запрос состоит более чем из одного слова, то наиболее адекватной будет аннотация, в которой встретились полностью все слова запроса, а не большое количество повторений отдельных слов запроса. То есть, наиболее существенной будет считаться аннотация, $Sum(W_i)$ которой максимальна.

2. Слова любого текста в информационном плане неравнозначны. Из этого следует, что вместо числа *n* слов запроса целесообразнее использовать сумму весов каждого встретившегося слова запроса.

3. Если ключевые слова и их веса рассчитываются для набора документов, объединенных одной темой, то важность слова для рубрики и для каждого конкретного документа различается. В связи с этим в весе слова запроса необходимо учитывать частоту его встречаемости в каждом документе.

Исходя из вышесказанного, кроме слов запроса следует учитывать другие слова документа, которые имеют высокую частоту встречаемости. Это положение учитывается в алгоритме *Freq*, реагирующем на частоту слов в окне длиной в 1000 слов вокруг слова запроса (анализируемый фрагмент находится в середине окна). При этом отбирается некоторое количество слов, которые встречаются в данном фрагменте наиболее часто. Для вычисления веса слова используется формула $W_{freq} = W_b + Sum(log2(F_k))$, где W_b — вес, вычисленный по базовому алгоритму, F_k — частота слова в окне в 1000 слов, включающем фрагмент. Наибольший вес получают те фрагменты, которые содержат наибольшее число слов запроса, а также большее количество часто встречающихся в найденном документе слов. Однако частота встречаемости слова в документе не несет информации о его точном распределении по тексту (распределено ли оно равномерно по всему документу или только в некоторых фрагментах). Кроме того, вычисление частот слов для фрагментов документов может занять много времени и ресурсов.

В последнее время определенную популярность приобрели статистические методы, основанные на Марковских цепях. Однако они еще более сложны. Поэтому довольно распространенным является алгоритм *LRU-K*, представляющий собой вариант алгоритма «последний недавно использованный». Алгоритм *LRU-K* состоит из следующих этапов.

1. При инициализации создаются три структуры данных: массив слов и два массива с указателями на слова (*array1* и *array2*) и длинами *k*.

2. При обработке документа для каждого слова производится поиск в массиве слов. Если слово не найдено, то ссылка на него помещается в массив *array1* в первую позицию. Остальные позиции в массиве сдвигаются, самое последнее слово удаляется из *array1* и из массива слов. Если слово найдено и встречается в массиве *array1*, то оно из него удаляется и переносится на первую позицию в массив *array2*. При этом если массив *array2* полностью заполнен, то последнее слово из него также удаляется, как и в первом случае. Если слово найдено и уже было в массиве *array2*, то оно просто перемещается на первую позицию.

Если бы слова в тексте имели равную вероятность появления, то после обработки фрагмента текста, содержащего слов намного больше *k*, содержимое массива *array2* совпадало бы с *k* наиболее часто встретившихся слов. То есть данный алгоритм можно рассматривать как один из вариантов оценки локальной частоты терминов при предположении равномерного распределения слов. Однако, предлагаемый алгоритм, кроме этого, выделяет слова, которые имеют не только высокую частоту, но и равномерно распределены вблизи выбираемо-

го фрагмента. Вычисление веса фрагмента найденного документа производится также как и для алгоритма *Freq*. Но вместо суммы, вычисляемой по наиболее часто встречающимся словам, к весу, определенному по базовому алгоритму, прибавляется количество слов из массива *array2*, встретившихся в анализируемом фрагменте.

ЛИТЕРАТУРА

1. Браславский, П. Автоматическое реферирование Веб-документов с учетом запроса / Автоматическая обработка данных. Интернет математика. — М., 2005. — С. 485–499.

2. Губин, М.В. Эффективный алгоритм формирования контекстно-зависимых аннотаций. [Электронный ресурс]. — Режим доступа: http://www/dialog-21.ru/archive/2005/Gubin%20Merkulov/GubinM_MerkulovA.htm. — Дата доступа: 12.03.2013.

3. Вороной, С.М. Повышение эффективности интеллектуального поиска в полнотекстовых базах данных на основе автоматического аннотирования документов. [Электронный ресурс]. — Режим доступа: http://iai.donetsk.ua/public/journalAI_2005_3/razdel14/02_Babin_Voronoy_ Malashchuk.pdf. — Дата доступа: 12.03.2013.