Инженерный вестник Дона, №1 (2015)
ivdon.ru/ru/magazine/archive/n1y2015/5877

Модель поиска смысла текстовых фрагментов

А.И. Джангаров, Х. А. Ахметова

Чеченский государственный университет, г. Грозный

Аннотация: Объектом исследования данной научной работы является рассмотрение и анализ такой области, как информационный поиск. Учитывая стремительный рост количества электронных документов, становятся очень актуальными задачи их кластеризации, классификации и упорядочивания. В работе также указывается востребованность в создании формальных методов семантики текстов и их обработки. Помимо этого, был создан программный продукт, реализующий такой функционал семантического анализа, как контекст и контекстная связка. Научная новизна состоит в том, чтобы научить машину раскрывать физический смысл в документации технического характера, для оптимизации поиска необходимых данных.
Ключевые слова: семантика, контекстная связка, обратная польская запись, кластеризация, контекст.

На данный момент довольно актуальной является задача оптимизации обработки текстовой информации. Ежедневно возрастает количество различных интернет-ресурсов и сайтов, содержащих огромное количество информации. В рамках данной научно-исследовательской работы, главным образом, рассматривается и анализируется текстовая информация [1].

В связи с этим было решено обратить внимание на такой раздел лингвистики, как семантика. Семантика как раз таки позволяет анализировать текст и выделять в нем смысл. Однако, стоит отметить, что семантический анализ справляется, в основном, только с текстами, которые носят технический характер. Задача распознавания смысла, например, в художественной литературе является до сих пор недостижимой. Связано это с наличием литературных оборотов и приемов, которые несут в себе скрытый или двойной смысл [2].

Актуальность исследования

На сегодняшний день, вся информация, которую пользователь может найти в сети, хранится на специальных физических и облачных серверах. Если рассмотреть частный случай доступа к информации при помощи поисковых систем, то главным образом информация хранится в так называемых дата-центрах (рисунок 1).

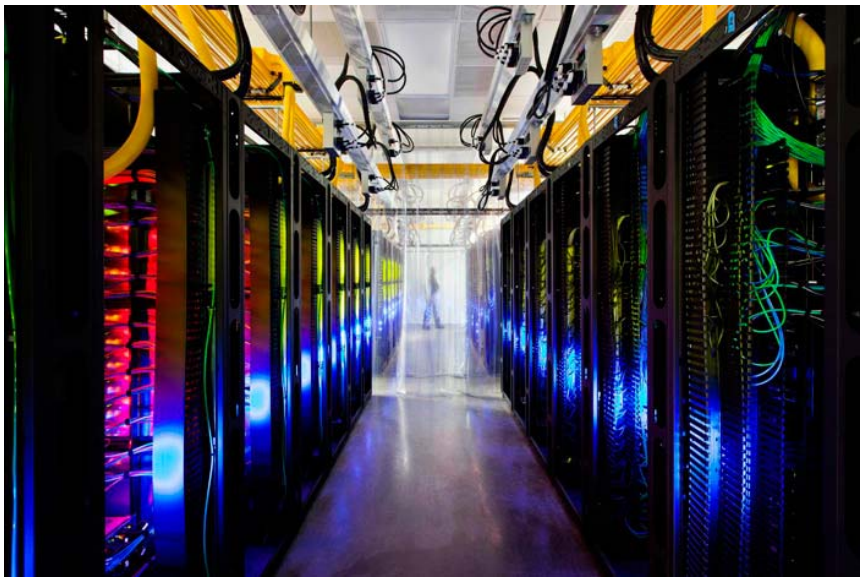

Рис. 1. – Дата центр компании Google

Для того, чтобы предоставить пользователю интересующую его страницу, была разработана программа под названием «веб-паук» или «веб-краулер». Это программа, перебирающая страницы Интернета и сохраняющая данные о них в базу поисковой системы. Для того, чтобы процесс поиска необходимой информации занимал сравнительно небольшое количество времени, а также был точным по смыслу, в большинстве случаев, применяются инструменты и механизмы семантического анализа. С учетом того, что объемы информации стремительно увеличиваются день за днем, существует высокая актуальность и востребованность задачи автоматизации обработки и поиска необходимой текстовой информации [3].

Метод реализации

Для решения поставленной задачи было решено использовать такую форму записи выражений, как обратная польская запись. Данная запись была разработана австралийским философом Чарльзом Хэмблином. Отличительной чертой этой записи является то, что операнды выражений располагаются перед знаками операций [4].

В одном из научных трудов Р.Ю. Вишнякова и Ю.М. Вишнякова приводится ряд важных свойств и понятий обратной польской записи:

1. Смысловой функционал (совокупность уникальных операндов выражения) текстового фрагмента $\alpha = x_1, x_2, ..., x_k$ представляется в следующем виде:

$$S(\alpha) = \Phi\ (S(x_1),\ S(x_2), ..., S(x_n));$$

$$S(\alpha) \subset S(x_i),$$

где $x_1, x_2, ..., x_k$ – слова текстового фрагмента α, который представляет собой интерпретацию предложения, x_i – главное слово в выражении α, а $\Phi\ (S(x_1),\ S(x_2), ..., S(x_n))$ – смысловой функционал.

2. При наличии в выражении слов x (главное слово) и y (зависимое слово), уточнение контекстного смысла $\vec{\cap}$ выполняется следующим образом:

$$S\overrightarrow{(x:y)} = S(x)\ \vec{\cap}\ S(y),$$

$$S\overrightarrow{(x:y)} \subset S(x),$$

здесь: $S(x)$ и $S(y)$ определяют смысл слов x и y, $S\overrightarrow{(x:y)}$ – итоговый результат контекстного выделения смысла. Чтобы лучше понимать разницу между главными и зависимыми словами можно привести следующий пример: главное слово выделяет объект или совокупность объектов, а зависимое – характеристики и отличительные особенности этих объектов [5].

Обоснование выбора программной платформы реализации

При выборе программной среды реализации обратной польской записи основной уклон делался на наличие инструментальных средств, которые позволили бы:

1. создать понятный и удобный в использовании графический интерфейс;
2. иметь возможность создания собственных баз данных;
3. использовать CASE-технологии [6].

Кроме того, основными критериями являлись показатели производительности, ресурсоемкость создаваемого программного продукта и наличие большого количества необходимых библиотек.

В связи с этим было выделено две платформы, подходящие под описание: Visual C++ и Delphi 7. Итоговый выбор был сделан в пользу последней платформы, так как среда разработки Delphi выгодно отличается наличием большого числа различных компонентов и библиотек, что ощутимо влияет на простоту и скорость разработки [7].

Программная реализация

Для программной реализации обратной польской записи было решено воспользоваться алгоритмом Дейкстры. Для этого, исследуемые записи представляются в виде корневого графа, узлы которого (*x, y, z*) являются словами, а вершина *v* главным словом (рисунок 2) [8]:

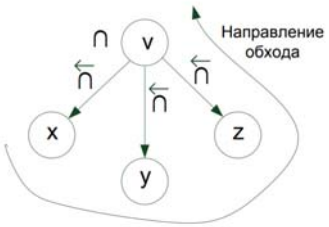

Рис. 2. – Дерево контекстной связки

Алгоритм обхода графа и построение обратной польской записи состоит из следующих шагов:

1. На первом шаге происходит выбор стартовой позиции в графе и начинается его обход слева направо.
2. На втором шаге формируется обратная польская запись всех узлов графа, с целью уточнения контекстного смысла и поиска главного и зависимого слова [9].

При обработке программой предложения, на выходе пользователь получает словосочетания, имеющие контекстную связку и информацию о том, какое слово является главным, а какое зависимым (рисунок 3):

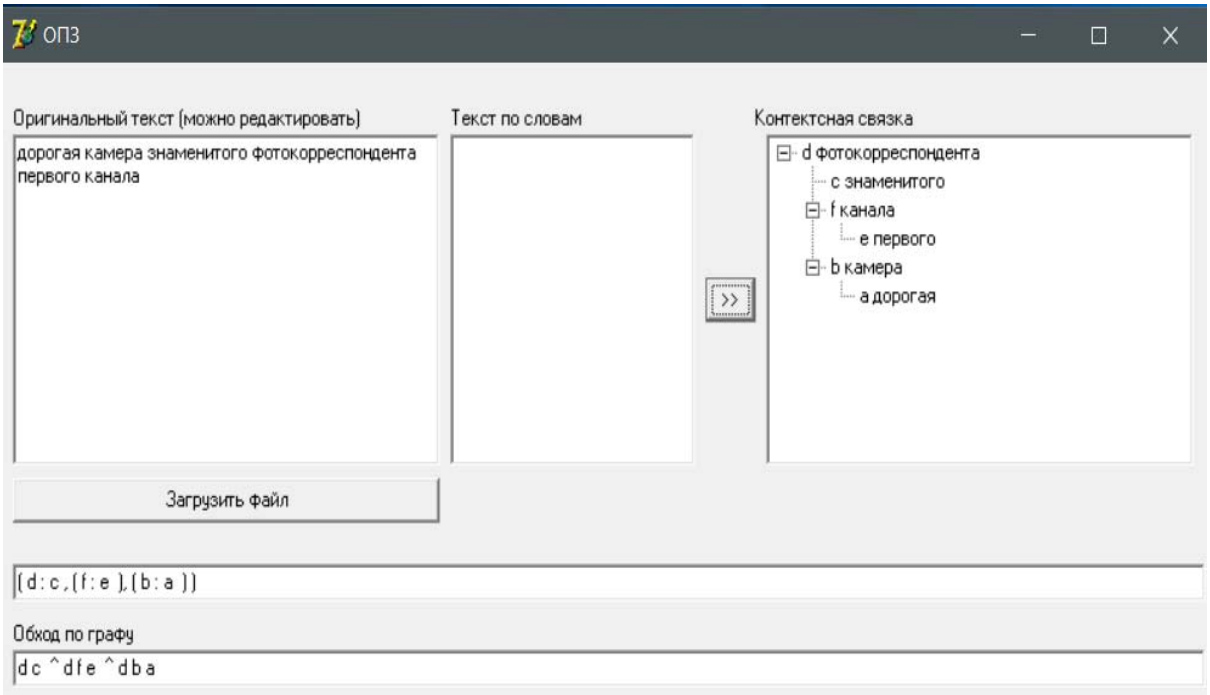

Рис. 3. – Программная реализация

Также строится обратная польская запись, которая демонстрирует порядок обхода узлов графа. Алгоритм Дейкстры завершается, если были инициализированы все узлы. Главная сложность заключалась в том, что при обработке текстовой информации отсутствовали операции между

операндами выражения в явном виде. Поэтому для создания обратной польской записи выражений использовались операции конкатенации и разделения текстовых строк (выделение узлов). Повторные испытания подтвердили корректную работу программы (рисунок 4):

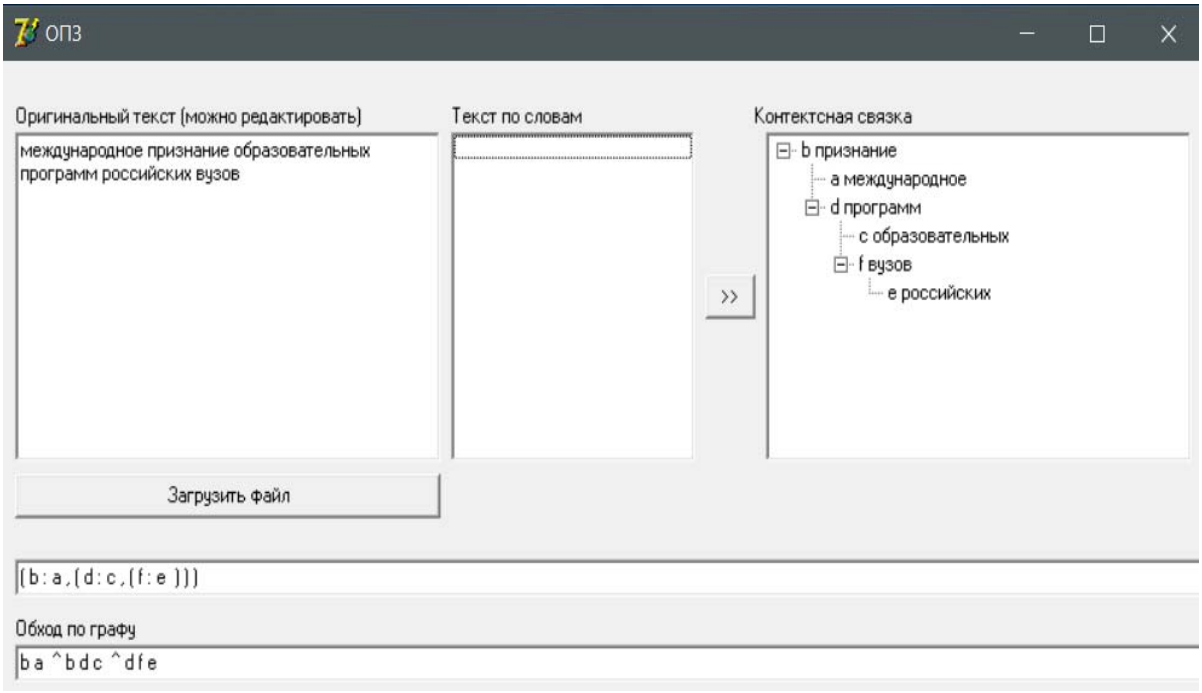

Рис. 4. – Дополнительный пример программной реализации

Заключение

В работе были рассмотрены основные и наиболее важные моменты, касающиеся представления смыслового функционала текстовых записей в виде обратной польской записи. Для определения контекстной связки было предложено использовать графы и алгоритм обхода Дейкстры. Показано, что основной особенностью данной записи является сохранение однопроходного линейного порядка вычислений. Также был создан программный продукт в среде программирования Delphi 7, главной целью которого является автоматизация процесса поиска главных и зависимых слов в тексте и создание контекстных связок [10].

Литература

1. Dodin Jean-Daniel. Enter: Reverse Polish Notation Made Easy. SYNTHETIX. 2012. 146 p.
2. Gieck Kurt, Gieck Reiner. Engineering Formulas. McGraw Hill Professional. 2015. 572 p.
3. Fuchs Marjorie, Bonner Margaret Focus on Grammar: An Integrated Skills Approach. SYNTHETIX. 2014. 224 p.
4. Nelson David The Penguin Dictionary of Mathematics: Fourth edition. Penguin UK. 2018. 496 p.
5. Вишняков Р.Ю., Вишняков Ю.М. Вычислительное представление смысла текстовых фрагментов на основе обратной польской записи. Известия ЮФУ. Технические науки. 2013. 7 с.
6. Hassitt Anthony Computer Programming and Computer Systems. Print Replica. 2017. 321 p.
7. Magomedov I. A. Mezhieva A.I. Suleymanova M.A. Inženernyj vestnik Dona (Rus). 2018. №4. URL: ivdon.ru/ru/magazine/archive/n4y2018/5334
8. Bamberg Paul A Course in Mathematics for Students of Physics: Volume 1 (Bk. 1). NeatTechBooks. 2018. 187 p.
9. Heaton Jeff Introduction to the Math of Neural Networks. Heaton Research. 2016. 119 p.
10. Ball, John A. Algorithms for RPN (Reverse Polish Notation) Calculators. John Wiley & Sons Inc. 2016. 222 p.

References

1. Dodin Jean-Daniel. Enter: Reverse Polish Notation Made Easy. SYNTHETIX. 2012. 146 p.
2. Gieck Kurt, Gieck Reiner. Engineering Formulas. McGraw Hill Professional. 2015. 572 p.

3. Fuchs Marjorie, Bonner Margaret Focus on Grammar: An Integrated Skills Approach. SYNTHETIX. 2014. 224 p.
4. Nelson David The Penguin Dictionary of Mathematics: Fourth edition. Penguin UK. 2018. 496 p.
5. R.Y. Vishnyakov, Y.M. Vishnyakov. IZVESTIYa YuFU. TEXNIChESKIE NAUKI. 2013. 7 p.
6. Hassitt Anthony Computer Programming and Computer Systems. Print Replica. 2017. 321 p.
7. Magomedov I. A. Mezhieva A.I. Suleymanova M.A. Inženernyj vestnik Dona (Rus). 2018. №4. URL: ivdon.ru/ru/magazine/archive/n4y2018/5334
8. Bamberg Paul A Course in Mathematics for Students of Physics: Volume 1 (Bk. 1). NeatTechBooks. 2018. 187 p.
9. Heaton Jeff Introduction to the Math of Neural Networks. Heaton Research. 2016. 119 p.
10. Ball, John A. Algorithms for RPN (Reverse Polish Notation) Calculators. John Wiley & Sons Inc. 2016. 222 p.