КАЧЕСТВО ПРОГРАММНЫХ СРЕДСТВ КОМПЬЮТЕРНОЙ ЛИНГВИСТИКИ

В. В. Бахтизин, А. В. Макаренко

Белорусский государственный университет информатики и радиоэлектроники,
кафедра программного обеспечения информационных технологий
ул. П. Бровки, 6, г. Минск, Республика Беларусь
телефон: + (375 29) 35 000 31; e-mail: makarenko_adam@mail.ru
web: www.bsuir.by

Данная работа посвящена актуальным проблемам повышения качества программных средств компьютерной лингвистики. В статье затрагиваются такие характеристики качества программных средств как функциональность и эффективность.

Ключевые слова – качество программных средств, корпусная лингвистика, частотный словарь.

1 ФОРМИРОВАНИЕ ЧАСТОТНОГО СЛОВАРЯ

В настоящее время вопрос повышения качества программных средств является одним из наиболее приоритетных, так как его успешное решение даёт преимущество успешно конкурировать на рынке ПС. Особого внимания заслуживает вопрос повышения качества ПС в таком направлении искусственного интеллекта, как компьютерная лингвистика. Благодаря этому можно получить более точные выходные данные анализа частотных свойств текста и лингвистических корпусов.

Если результат работы ПС компьютерной лингвистики зависит от частоты встречаемости отдельных лексем, то одним из шагов роста качества этого ПС является увеличение качества формирования частотного словаря.

Частотный словарь (или частотный список) – это набор слов какого-либо языка или подъязыка вместе с информацией о частоте их встречаемости.

На рисунке 1 изображена схема формирования частотного словаря. Частотный анализатор позволяет определить для каждого слова s_i из системного словаря $S=\{s_i \mid i=\overline{1,n}\}$ его частоту вхождения f_i в данный корпус $K=\{k_j \mid j=\overline{1,m}\}$, где k_j – слово, извлечённое из текстов, принадлежащих корпусу K. Частотная характеристика – это множество $F=\{f_i \mid i=\overline{1,n}\}$, мощность которого равна количеству слов n в системном словаре S. В свою очередь корпус состоит из текстов, составленных в соответствии с определёнными принципами, размеченных по определённому стандарту и обеспеченных специализированной поисковой системой. Например, корпусом можно считать собрание текстов, объединённых каким-то общим признаком (стилем, жанром, автором, периодом создания текстов, языком).

Системный словарь – это полный набор лексем (или словоформ) данного языка или подъязыка.

$$f_i=\sum_{j=1}^{m} e(s_i,k_j), \tag{1}$$

$$\text{где } e(s_i,k_j)=\begin{cases}0, \text{ если } s_i \neq k_j\\ 1, \text{ если } s_i = k_j\end{cases}. \tag{2}$$

Таким образом, для каждого слова s_i определяется число его вхождений $f_i \geq 0$ в данный корпус.

Рис. 1. Схема формирования частотного словаря

2 МОДЕЛЬ КАЧЕСТВА ПРОГРАММНЫХ СРЕДСТВ ФОРМИРОВАНИЯ ЧАСТОТНОГО СЛОВАРЯ

В данной работе затрагиваются такие характеристики качества программных средств как функциональность и эффективность.

Функциональность – способность программного продукта обеспечивать функции, удовлетворяющие установленные и подразумеваемые потребности при применении ПС в заданных условиях. Одной из подхарактеристик функциональности является правильность. Правильность – способность программного продукта обеспечивать правильные или приемлемые результаты или эффекты с необходимой точностью [2].

Так как все результаты частотного анализатора носят вероятностный характер, следовательно, нельзя требовать, чтобы программный продукт обеспечивал абсолютно правильные результаты. Чтобы добиться приемлемых результатов, можно расширить внутреннюю метрику «Точность» из стандарта ISO/IEC TR 9126-3:2003 [3] с учётом первого закона Зипфа [4]:

$$C = \frac{f_i * r}{n}, \quad (4)$$

где C – константа Зипфа; f_i – частота вхождения слова в корпус K; r – ранг частоты; n – количество слов в системном словаре S.

Метрику можно записать следующим образом:

$$X = \frac{A}{B} \ (0 < X \le 1), \quad (5)$$

где A – максимальное количество слов в частотном словаре, для которых константа Зипфа будет одинакова; B – общее количество слов в системном словаре.

Пример 1. Пусть системный словарь S состоит из четырёх слов (n=4). Вычислим C для каждого i-го слова.

Таблица 1

i	1	2	3	4
r	1	2	3	4
s_i	и	в	не	он
f_i	120	60	40	30
Ci	30	30	30	30

В данном примере C=30 для каждого s_i. A=4, B=n=4. Следовательно, $X = \frac{A}{B} = \frac{4}{4} = 1$.

Пример 2. Пусть системный словарь S состоит из пяти слов (n=5). Вычислим C для каждого i-го слова.

Таблица 2

i	1	2	3	4	5
r	1	2	3	4	5
s_i	и	в	не	он	на
f_i	120	65	40	25	24
Ci	24	26	24	20	24

В данном примере C=26 для одного слова, C=24 для трёх слов и C=20 для одного слова. A=3, B=n=5. Следовательно, $X = \frac{A}{B} = \frac{3}{5} = 0{,}6$.

С увеличением относительного значения метрики значение правильности должно увеличиваться. Следовательно, частотный словарь в примере 1 был сформирован более правильно, чем частотный словарь в примере 2. Наиболее правильные результаты будут достигаться, если все элементы частотного словаря будут удовлетворять первому закону Зипфа (X=1).

Одной из характеристик качества программного обеспечения является эффективность. Эффективность – способность программного продукта обеспечить соответствующую производительность в зависимости от количества используемых вычислительных ресурсов в заданных условиях [2].

Чтобы повысить эффективность частотного анализатора, необходимо реализовать эффективный метод поиска слова в системном словаре. Для поиска слова из корпуса K в системном словаре S существует большое количество алгоритмов.

В данной работе предлагается усовершенствовать метод последовательного поиска. Данный метод осуществляет последовательное считывание данных с системного словаря S и сравнивает s_i с k_j. Зипф эмпирически на основании анализа произвольных англоязычных текстов заметил закономерность – «слова с большим количеством букв встречаются в тексте реже коротких слов» [4]. Следовательно, когда элементы системного словаря отсортированы по возрастанию частоты, можно увеличить скорость последовательного поиска, если начинать поиск с конца системного словаря для длинных слов и с начала системного словаря для коротких слов.

Основным достоинством этого метода является простота реализации и, при условии невысокой фрагментации на диске, достаточно высокая скорость работы, так как в этом случае при последовательном считывании достигается пиковая скорость чтения. Также метод позволяет реализовать многофункциональный поиск без существенных ограничений (например, по подстроке или по регулярным выражениям).

Предложенная модель качества позволяет предсказать будет ли разрабатываемый программный продукт удовлетворять требованиям к правильности результатов во время тестирования или эксплуатации [2], а также позволяет увеличить эффективность формирования частотного словаря при использовании метода последовательного поиска. Примером программных средств, для которых можно использовать данную модель качества, могут служить архиваторы, поисковые системы, программы-консультанты, программы-справочники, приложения компьютерной лингвистики, а также программные средства, которые используются для преподавания иностранных языков, для создания новых словарей, текстов и книг.

ЛИТЕРАТУРА

[1] Пиотровский, Р. Г. Математическая лингвистика / Р. Г. Пиотровский, К. Б. Бектаев, А. А. Пиотровская. – М. : Высшая школа, 1972. – 383 с.

[2] Бахтизин, В.В. Стандартизация и сертификация программного обеспечения : учебное пособие / В. В. Бахтизин, Л. А. Глухова. – Минск : БГУИР, 2006. – 200 с.

[3] ISO/IEC TR 9126-3:2003. Программная инженерия – Качество продукта – Часть 3: Внутренние метрики.

[4] Чалей, И.В. Исследование применимости законов Зипфа к русскоязычным текстам / И. В. Чалей, Э. Р. Гасанов, Н. В. Лисицын. – СПб, 2007. – 16 с.