УДК 681.3

И.А. Андреев, В.А. Башаев, В.В. Клейн, Н.Г. Ярушкина

КОМБИНИРОВАНИЕ СТАТИСТИЧЕСКОГО И ЛИНГВИСТИЧЕСКОГО МЕТОДОВ ДЛЯ ИЗВЛЕЧЕНИЯ ДВУХСЛОВНЫХ ТЕРМИНОВ ИЗ ТЕКСТА

Андреев Илья Алексеевич, *студент факультета информационных систем и технологий Ульяновского государственного технического университета. Опубликовано несколько статей в области извлечения информации из текста. [e-mail: ares-ilya@yandex.ru].*

Башаев Виталий Александрович, *аспирант, окончил факультет лингвистики и международного сотрудничества Ульяновского государственного университета. Имеет статьи в области извлечения информации из текста. [e-mail: perevod73@yandex.ru].*

Клейн Виктор Викторович, *студент факультета информационных систем и технологий УлГТУ. Опубликовано несколько статей в области извлечения информации из текста. [e-mail: vikklein93@gmail.com].*

Ярушкина Надежда Глебовна, *доктор технических наук, профессор, заведующий кафедрой «Информационные системы» УлГТУ. Имеет более 250 научных работ в области мягких вычислений, нечеткой логики, гибридных систем. [e-mail: jng@ulstu.ru].*

Аннотация

В данной статье рассматриваются результаты эксперимента по комбинированию лингвистического и статистического методов для извлечения двухсловных терминов из текста по предметной области «Станки с числовым программным управлением». Помимо эксперимента и результатов, особое внимание уделено описанию архитектуры разработанного программного обеспечения.

Ключевые слова: извлечение терминов, лингвистический метод, статистический метод, частота биграмм.

Ilia Alekseevich Andreev, *a student of the Faculty of Information Systems and Technologies at Ulyanovsk State Technical University; an author of several articles in the field of information extraction from a text. e-mail: ares-ilya@yandex.ru.*

Vitalii Aleksandrovich Bashaev, *a post-graduate student, graduated from the Faculty of Linguistics and International Cooperation at Ulyanovsk State University; an author of articles in the field of information extraction from a text. e-mail: perevod73@yandex.ru.*

Viktor Viktorovich Klein, *a student of the Faculty of Information Systems and Technologies at Ulyanovsk State Technical University; an author of several articles in the field of information extraction from a text. e-mail: vikklein93@gmail.com.*

Nadezhda Glebovna Iarushkina, *Doctor of Engineering, Professor, Head of the Department of Information Systems at Ulyanovsk State Technical University; an author of more than 250 papers in the field of soft computation, fuzzy logic, and hybrid systems. e-mail: jng@ulstu.ru.*

Abstract

This article contains the experiment results on the combination of linguistic and statistical methods for extraction of two-word terms from a text on "CNC Machines" discipline. Along with the experiment and the results, a special attention is paid to the description of the developed software architecture.

Key words: term extraction, linguistic method, statistic method, bigrams frequency.

Введение

Задача извлечения терминологии из текста возникает в библиотечном деле, лексикографии и терминоведении. Прикладные аспекты автоматизированного извлечения терминов из текста имеются в информационном поиске и машинном обучении. Под «извлечением терминологии» (term extraction) понимается обработка текста на определенном языке и формирование списка терминов-кандидатов для добавления в словарную базу. Эта операция позволяет избавиться от терминологической избыточности и добиться последовательности терминологии, а потому имеет большое значение в управлении терминологией (terminology management) в переводческих задачах и задачах САПР при необходимости анализа больших массивов документации и унификации используемой терминологии.

Принцип работы существующих алгоритмов извлечения терминологии основан на статистических и лингвистических методах. В основе статистических методов лежит вычисление степени терминологичности на основании числовых закономерностей, присущих термину или нетермину. В основе лингвистических методов лежит отбор по определенным лексико-грамматическим шаблонам и другим лингвистическим признакам термина [1].

Алгоритмы извлечения, которые основаны только на статистических методах, универсальны для разных языков, а алгоритмы, которые основаны на лингвистическом анализе, более сложны и ограничены конкретным языком. Однако, как показано в ряде публикаций, качество извлечения выше у алгоритмов, которые основаны на комбинировании признаков и включают лингвистический анализ [2].

При работе с большими предметными областями и корпусами даже лучшие методы извлечения терминологических словосочетаний показывают значительное падение процентного содержания терминов с 90% на первой сотне списка извлеченных терминологических словосочетаний до 60% на третьей тысяче [3]. Таким образом, актуальной является разработка новых алгоритмов и методов по улучшению качества упорядочения списков словосочетаний с целью повышения процентной доли терминов в начале списка.

1 Теоретические модели

1.1 Frequency

Применение статистических методов опирается на представление о том, что термины, как правило, это наиболее частотные слова и словосочетания, встречающиеся в специальных текстах и выражающие понятия предметной области. Терминосочетания обычно соотносятся с n-граммами (двух-, трех-, четырехчленными сочетаниями), характеризуются высокой степенью устойчивости. В качестве мер, пригодных для оценки устойчивости словосочетаний в специальных текстах, следует упомянуть Frequency, MI (Mutual Information), T-score, Log-Likelihood, C-value, критерий χ^2 и ряд других.

В обсуждаемом эксперименте применяется метод Frequency, или метод прямого подсчета частот двусловий:

f_{xy}, где f_{xy} – частота биграммы xy.

Выполняется подсчет абсолютных частот всех двусловий (частоты встречаемости отдельных слов не учитываются) и простое упорядочивание списка на основании частоты, начиная с большей. Отбираются все пары слов, не разделенные знаками препинания (кроме дефиса). Метод Frequency исходит из того, что высокочастотные двусловия обозначают значимые понятия текста. Этот метод является одним из самых значимых методов извлечения терминологии [4] и [5].

1.2 T-Score

$$\frac{f_{xy} - \frac{f_x f_y}{n}}{f_{xy}^2},$$

где f_{xy} – частота биграммы,

f_x, f_y – частота x и y соответственно;

n – количество биграмм в корпусе.

Метод T-score предназначен для вычисления степени взаимосвязи двух слов. Мера T-score представляет собой степень доверия, с которой можно утверждать, что между двумя словами имеется определенная связь. Является несколько модифицированным ранжированием двусловий по частоте. Очевидно, что значение данной меры тем выше, чем выше частота двусловия в коллекции.

Цель метода T-score заключается в приближении биномиального распределения дискретной случайной величины к распределению, близкому к нормальному распределению непрерывной случайной величины N с опорой на нулевую гипотезу о независимости. Поскольку метод T-score является нормальной аппроксимацией биномиального распределения, он имеет те же известные недостатки, связанные с допущением о нормальном распределении.

1.3 Mutual Information (коэффициент взаимной информации)

$$MI = \log_2 \frac{f(x,y) \times N}{f(x) \times f(y)},$$

где $f(x,y)$ – частота биграммы; $f(x), f(y)$ – частота каждого слова в отдельности; N – количество слов в корпусе.

Мера MI вычисляет вероятность двух встречающихся вместе слов путем сравнивания произведения их относительных частот в корпусе с наблюдаемыми частотами их совместной встречаемости. Разница между этими величинами выявит степень значимости их встречаемости.

Если значение $MI(x; y)$ больше 1, тогда данное сочетание слов считается статистически значимым. В случае если $MI(x; y)$ примерно равно 0, сочетание слов является менее статистически значимым, слова появляются в паре крайне редко. $MI(x; y)$ меньше 0 означает, что x и y находятся в отношении дополнительной дистрибуции.

*1.4 Частотность с учетом частоты употребления в объемлющей коллекции (TF*IDF)*

TF (Term Frequency – частота слова) – численное значение, равное отношению количества вхождений какого-либо слова в документ к числу слов в нем. IDF (Inverse Document Frequency – обратная частота документа) – инверсия частоты встречаемости слова в документах коллекции или корпуса. Данный признак широко употребляется в информационно-поисковых системах и позволяет снижать вес употребительных слов.

$$TF * IDF(w) = TF * \log\left((n-b)/b\right),$$

где TF – это частотность слова в текущей коллекции;

n – размер контрастной коллекции;

b – число документов, в которых употреблялось слово w в контрастной коллекции.

1.5 Log-Likelihood

Log-Likelihood вводится с целью решения проблемы оценок методов T-Score и MI. В предположении о биноми-

альном характере функции распределения совместной встречаемости слов [6]

$$loglike = a * \log(a+1) + b * \log(b+1) + c * \log(c+1) + d * \log(d+1) - (a+b) * \log(a+b+1) - (a+c) * \log(a+c+1) - (b+d) * \log(b+d+1) - (c+d) * \log(c+d+1) + (a+b+c+d) * \log(a+b+c+d+1),$$

где a – частотность данного словосочетания (пары);

b – суммарная частотность других (отличных от данной) пар с той же самой левой леммой;

c – суммарная частотность других пар с той же самой правой леммой;

d – суммарная частотность пар, отличных от данной и не попадающих в категории (b) и (c).

1.6 C-Value

$$C\text{-}Value(a) = \begin{cases} \log_2 |a| * freq(a), \text{если не вложен}, \\ \log_2 |a| * freq(a) - \frac{1}{P(T_a)} * \sum_{b \in T_a} freq(b), \end{cases}$$

где a – кандидат в термины;

$|a|$ – длина словосочетания, измеряемая в количестве слов;

$freq(a)$ – частотность a;

T_a – множество словосочетаний, которые содержат a;

$P(T_a)$ – количество словосочетаний, содержащих a.

C-Value представляет собой независимый от предметной области метод автоматического извлечения многословных вложенных терминов (терминологическое словосочетание, которое является частью более длинного терминологического словосочетания). Метод C-Value использует данные лингвистического анализа (части речи, лингвистические шаблоны для отделения определенного типа терминов, стоп-лист) наряду со статистическим анализом.

1.7 Лингвистический метод извлечения терминов

В основе этого метода лежит отбор слов и словосочетаний согласно лексико-грамматическим шаблонам, присущим термину. Такими шаблонами являются: N, $N+N$, $N+N+N$, $A+N$, $A+A+N$ (N – существительное, A – прилагательное) и другие. Соответственно, при автоматизированном способе отбору предшествует морфологический анализ текста, заключающийся в определении частей речи, морфологических признаков, канонических форм словоупотреблений и другой информации.

1.8 Комбинированные методы

Комбинированные методы анализа терминологии предполагают совместное использование аппарата лексико-грамматических шаблонов, методов сборки терминосочетаний, системы фильтров, а также статистического аппарата. Каждый метод имеет свои недостатки, и комбинирование используется для исключения некоторых из них. Так, метод Frequency не учитывает связи между словами, что выражается в отборе в потенциальные термины высокочастотных двусловий, которые не являются терминами. Предлагаемый нами метод комбинирования позволяет отсечь такие словосочетания и повысить таким образом качество извлечения терминов.

Мы используем для анализа двусловия с частотой 2 и выше. Для фильтрации мы предлагаем использовать лексико-грамматические шаблоны, характерные для двухсловных терминов: Прил. + Сущ., Прич. + Сущ., Сущ. + Сущ. Этап морфологической разметки текста выполняет программа Mystem компании Яндекс [7].

Для проведения эксперимента мы использовали текст из руководства по эксплуатации токарно-фрезерного станка с числовым программным управлением (ЧПУ). Руководства по эксплуатации к станкам с ЧПУ содержат следующую информацию: конструкцию и принцип работы всех основных узлов и систем станка; правила техники безопасности; инструкции по эксплуатации, монтажу и пуско-наладочным работам; описание работ по техническому обслуживанию, диагностике и устранению неисправностей. К особенностям текстов данной предметной области можно отнести высокую насыщенность терминами, влияние научного стиля на лексико-семантические, морфологические, синтаксические параметры и формализованность содержания, опирающегося на логико-понятийную схему предметной области.

Объем текста составляет 14867 словоупотреблений. Эксперт по станкам с ЧПУ прочитал этот текст и выделил все терминологические единицы, которые имеют прямое отношение к предметной области. В результате мы получили эталонный список терминов количеством 373 термина (792 терминоупотребления).

1.9 Модель байесовского классификатора

В ходе дальнейшей разработки программы классификации планируется написание наивного байесовского классификатора, условная вероятная модель которого выглядит следующим образом:

$$P(H|E_1, E_2, E_3, E_4, E_5, E_6, E_7),$$

где H – гипотеза о том, является ли данное рассматриваемое слово или сочетание слов термином; E_1 – значение, основанное на показаниях реализованного в программе классификации лингвистического метода; E_2 – значение, основанное на результате, подсчитанном для данного словосочетания реализованным в программе классификации статистическим методом Mutual Information (актуально для биграмм); E_3 – значение, основанное на результате подсчета величины T-score для данного словосочетания (актуально для биграмм); E_4 – значение, основанное на результате подсчета величины TF*IDF (актуально для одиночных слов); E_5 – значение, основанное на результате подсчета величины Log-Likelihood (актуально для биграмм); E_6 – значение, основанное на результате подсчета величины C-Value (актуально для биграмм); E_7 – значение, основанное на данных, полученных в результате анализа онтологии предметной области.

При вычислении $P(H|E_1, E_2, E_3, E_4, E_5, E_6, E_7)$ нужно перейти к косвенным вероятностям, воспользовавшись теоремой Байеса:

$$P\left(H\left|E_1,E_2,E_3,E_4,E_5,E_6,E_7\right.\right)=$$
$$=\frac{P\left(E_1,E_2,E_3,E_4,E_5,E_6,E_7\left|H\right.\right)P\left(H\right)}{P\left(E_1,E_2,E_3,E_4,E_5,E_6,E_7\right)}.$$

Знаменатель можно исключить из рассмотрения, так как он представляет собой константу. Далее, допустив предположение о «наивности», то есть о независимости переменных $E_1, E_2, E_3, E_4, E_5, E_6, E_7$ друг от друга, распишем числитель следующим образом:

$$P(E_1,E_2,E_3,E_4,E_5,E_6,E_7\left|H\right.)P(H)=$$
$$=P(H)P(E_1,H)P(E_2,H,E_1)P(E_3,H,E_1,E_2)\times$$
$$\times P(E_4,H,E_1,E_2,E_3)P(E_5,H,E_1,E_2,E_3,E_4)\times$$
$$\times P(E_6\left|H,E_1,E_2,E_3,E_4,E_5\right.)\times$$
$$\times P(E_7\left|H,E_1,E_2,E_3,E_4,E_5,E_6\right.),$$

как следствие «наивности»:

$$P\left(H\right)P\left(E_1\left|H\right.\right)P\left(E_2\left|H\right.\right)P\left(E_3\left|H\right.\right)P\left(E_4\left|H\right.\right)\times$$
$$\times P\left(E_5\left|H\right.\right)P\left(E_6\left|H\right.\right)P\left(E_7\left|H\right.\right)=P\left(H\right)\prod_{i=1}^{7}P\left(E_i\left|H\right.\right).$$

По этой формуле рассчитывается вероятность для термина-кандидата.

Гипотеза о терминологичности данного слова или словосочетания зависит от разных оценок. Это означает, что следует описать 3 частные модели классификатора, основанные на общей модели. Так как не все используемые нами методы оценки в теории применимы для n-грамм с любым количеством слов, следует разделить модели на следующие виды: 1) модель для одиночных слов; 2) модель для биграмм; 3) модель для n-грамм ($n>2$ и $n<6$). Модель классификатора для одиночных слов выглядит таким образом: $P(H|E_1, E_4, E_7)$, где E_1 – значение, основанное на результате работы лингвистического метода; E_4 – значение, основанное на работе метода TF*IDF; E_7 – значение, основанное на анализе онтологии предметной области.

Таким образом, модель для одиночных слов включает в себя как значения «универсальных» методов – лингвистического и онтологического, так и специфического для одиночных слов TF*IDF-метода.

Модель классификатора для биграмм выглядит таким образом: $P(H|E_1, E_2, E_3, E_5, E_6, E_7)$, где E_1 – значение, основанное на результате работы лингвистического метода; E_2 – значение, основанное на результате, подсчитанном для данного словосочетания реализованным в программе классификации статистическим методом Mutual Information (актуально для биграмм); E_3 – значение, основанное на результате подсчета величины T-score для данного словосочетания (актуально для биграмм); E_5 – значение, основанное на результате подсчета величины Log-Likelihood (актуально для биграмм); E_6 – значение, основанное на результате подсчета величины C-Value; E_7 – значение, основанное на данных, полученных в результате анализа онтологии предметной области.

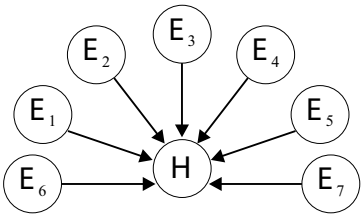
Таким образом, модель классификатора для биграмм включает в себя значения, полученные от «универсальных» методов и специфических методов для биграмм: MI, T-score, Log-Likelihood, C-Value.

Модель классификатора для n-грамм ($n>2$ и $n<6$) выглядит следующим образом: $P(H|E_1, E_7)$, где E_1 – значение, основанное на результате работы лингвистического метода; E_7 – значение, основанное на анализе онтологии предметной области.

В данную частную модель входят только значения от «универсальных» методов.

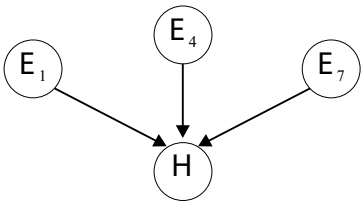
Вероятностные отношения для нашего классификатора представим в виде Байесовской сети – графической структуры, позволяющей представить распределение вероятностей над большим числом переменных. В нашем случае будет представлено 8 переменных для общей сети, представляющей распределение для общей модели классификатора, и соответственно, 3 переменные, 6 переменных и 2 переменные для одиночных слов, биграмм и n-грамм ($n>2$ и $n<6$). Следует отметить, что априорные вероятности E_n будут определены позднее в результате экспериментов.

Общая байесовская сеть для классификатора:

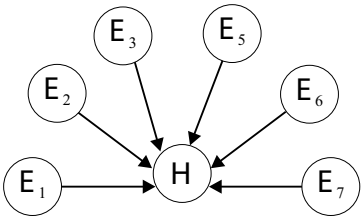


Данный ациклический граф представляет распределение вероятностей для используемой общей модели байесовского классификатора. Показана зависимость гипотезы от всех высчитываемых значений.

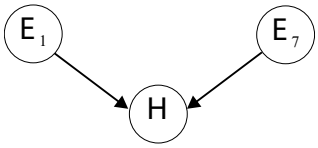
Рассмотрим сеть для классификатора одиночных слов:



Сеть для классификатора биграмм:



Сеть для классификатора n-грамм:



2 Программные средства

Для решения поставленной задачи была разработана база данных (БД), которая заполняется информацией из файла-выдачи программы Mystem. Далее происходит подсчет частоты всех двусловий и отсечение двусловий с частотой 1. После этого из полученного списка выделяются двусловия, удовлетворяющие значимым лексико-грамматическим классам Прил.+Сущ., Прич.+Сущ., Сущ.+Сущ.

2.1 Морфологический анализатор

В качестве морфологического анализатора русского языка мы используем программу Mystem компании Яндекс. На выходе мы имеем файл формата txt с размеченным текстом. Каждое словоупотребление начинается с новой строки, включая межсловные знаки (которые, кроме пробела, являются ограничителями потенциально устойчивого словосочетания). Для каждого словоупотребления выводятся лемма и набор признаков, обозначающих грамматические формы словоупотребления. Для нас представляют интерес следующие данные: словоупотребление, лемма, часть речи. Mystem для каждого словоупотребления выводит все возможные признаки с логическим «ИЛИ». Так, словоупотребление «Передняя» определяется как существительное ж. р., ед. ч, им. п. «Передняя» и как прилагательное м .р., ед. ч, им. п. «Передний».

2.2 Программа для классификации

Для классификации двусловий по частоте и лексико-грамматическим классам был разработан блок классификации данных морфологического анализа. Данный блок представляет собой программное средство для занесения в БД информации, получаемой от программы морфологического анализа текста Mystem, и модулей обращения к БД, работающих по определенным методам извлечения терминологии. Платформой для разработки данного программного обеспечения стала платформа Microsoft .NET, язык программирования C#. Для работы с БД используется сервер SQL.

Основные этапы разработки:

1. Модуль работы с морфологией Mystem;
2. Модуль заполнения БД;
3. Модуль графического представления БД;
4. Модуль осуществления запросов.

На диаграмме классов (рис. 1) показаны основные части программы, их взаимодействия, указаны методы и поля каждого класса. На основе классов-сущностей (части речи и двусловия) создаются объекты, заносимые в БД. Управляющие классы обеспечивают как работу с самой БД, так и возможность проведения экспериментов с применением лингвистического и статистических методов. Граничные классы обеспечивают интерфейс для пользователя и отображение содержимого БД.

БД представляет собой 5 несвязанных между собой таблиц (рис. 2). Столбцы таблиц для существительных, прилагательных и глаголов представляют собой характеристики соответствующих частей речи, лемму и слово в тексте, а также позицию слова в тексте и вхождение данного слова в какое-либо устойчивое словосочетание. Все служебные части речи заносятся в одну таблицу. Помимо леммы, слова в тексте, позиции слова в тексте и использованности слова отдельный столбец содержит название части речи этого слова. Таблица двусловий содержит само словосочетание и его частоту. Архитектура и ядро программы представлены на рисунках 3 и 4.

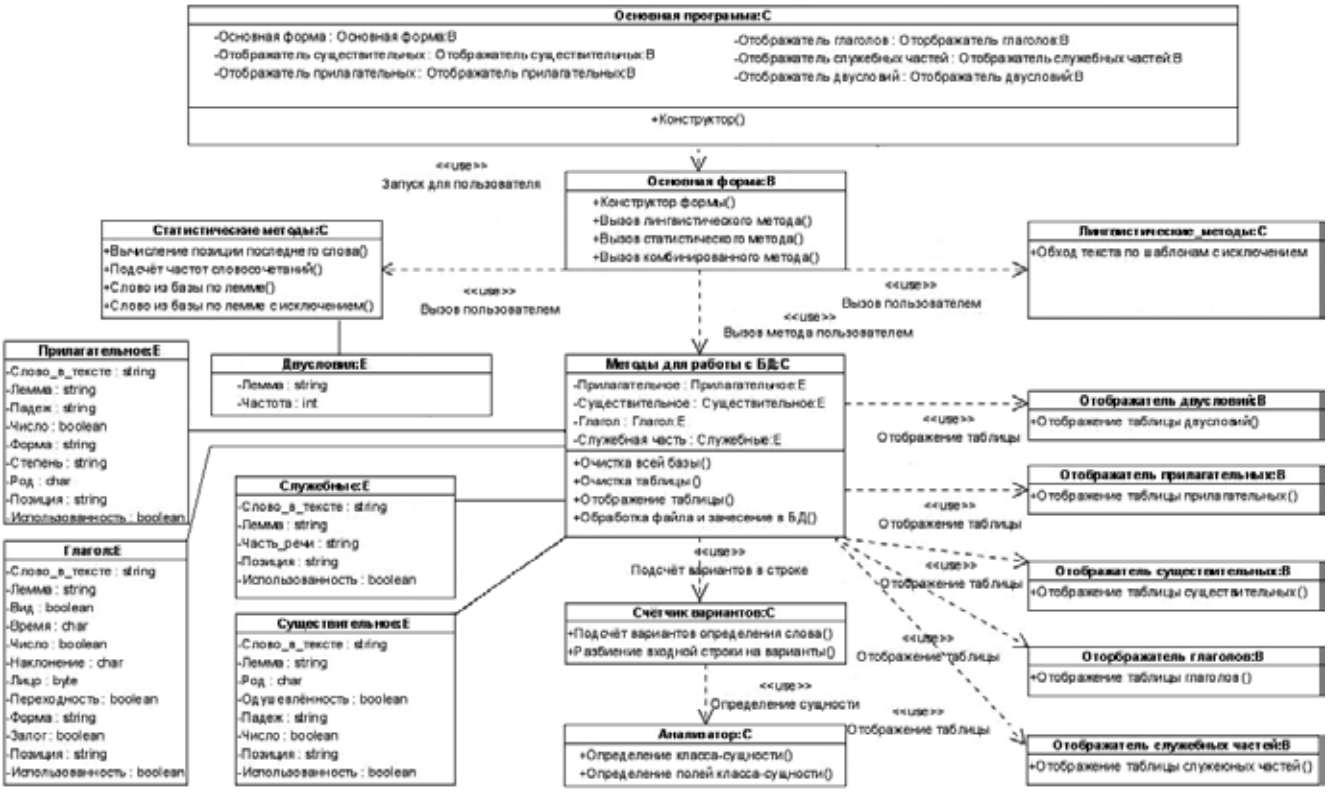


Рис. 1. Диаграмма классов

Существительное		
+ID	**integer(10)**	**Nullable = false**
Слово_в_тексте	varchar(255)	Nullable = true
Лемма	varchar(255)	Nullable = true
Род	char(1)	Nullable = true
Одушевлённость	bit	Nullable = true
Падеж	varchar(255)	Nullable = true
Число	bit	Nullable = true
Позиция	varchar(255)	Nullable = true
Использованность	bit	Nullable = true
TF	float(10)	Nullable = true
IDF	float(10)	Nullable = true

Прилагательное		
+ID	**integer(10)**	**Nullable = false**
Слово_в_тексте	varchar(255)	Nullable = true
Лемма	varchar(255)	Nullable = true
Падеж	varchar(255)	Nullable = true
Число	bit	Nullable = true
Форма	varchar(255)	Nullable = true
Степень	varchar(255)	Nullable = true
Род	char(1)	Nullable = true
Позиция	varchar(255)	Nullable = true
Использованность	bit	Nullable = true
TF	float(10)	Nullable = true
IDF	float(10)	Nullable = true

Глагол		
+ID	**integer(10)**	**Nullable = false**
Слово_в_тексте	varchar(255)	Nullable = true
Лемма	varchar(255)	Nullable = true
Вид	bit	Nullable = true
Время	varchar(255)	Nullable = true
Наклонение	varchar(255)	Nullable = true
Лицо	smallint(5)	Nullable = true
Переходность	bit	Nullable = true
Форма	varchar(255)	Nullable = true
Использованность	bit	Nullable = true
Позиция	varchar(255)	Nullable = true
TF	float(10)	Nullable = true
IDF	integer(10)	Nullable = true

Служебные части		
+ID	**integer(10)**	**Nullable = false**
Слово_в_тексте	varchar(255)	Nullable = true
Лемма	varchar(255)	Nullable = true
Часть речи	varchar(255)	Nullable = true
Позиция	varchar(255)	Nullable = true
Использованность	bit	Nullable = true
TF	float(10)	Nullable = true
IDF	float(10)	Nullable = true

Двусловия		
+ID	**integer(10)**	**Nullable = false**
Лемма	varchar(255)	Nullable = true
Частота	integer(10)	Nullable = true
tscore	float(10)	Nullable = true
MI	integer(10)	Nullable = true
ПерваяЧасть	varchar(255)	Nullable = true
ВтораяЧасть	varchar(255)	Nullable = true

Слово		
+ID	**integer(10)**	**Nullable = false**
Лемма	varchar(255)	Nullable = true
Частота	integer(10)	Nullable = true

Рис. 2. Entity-relationship (сущность-связь) диаграмма

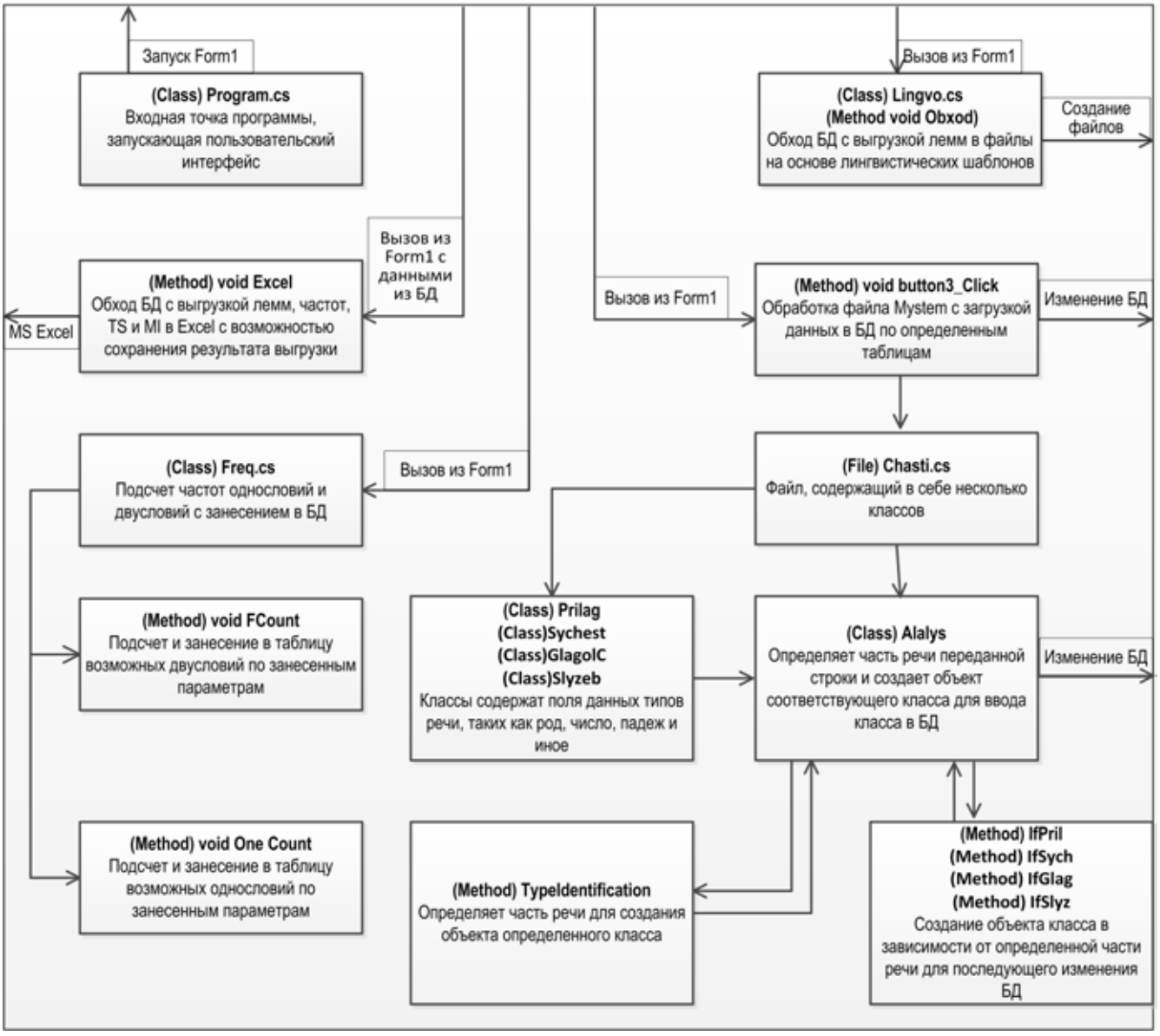


Рис. 3. Архитектура программы

2.3 Логика работы программы для подсчета частоты двусловий

Целью этого метода является подсчет частоты каждой комбинации из двух слов, встречающейся в тексте. Для этого в БД была добавлена отдельная таблица, хранящая леммы обоих слов и частоту самого двусловия. Подсчету частоты двусловий обязательно предшествует первичная обработка файла, сгенерированного Mystem.

Рис. 4. Ядро программы

Из всех четырех таблиц, содержащих морфологическую информацию о каждом слове из текста, определяется позиция последнего слова. Затем происходит перебор всех слов парами. Специальный метод осуществляет выдачу леммы из 4-х таблиц по позиции. Если этот метод возвращает значение null, это означает, что в тексте на этой позиции имел место разделитель, следовательно, двусловия как такового на этих двух позициях не существует – не осуществляется занесение двусловия в базу или подсчет его частоты.

Если по обоим позициям метод выдал какие-нибудь леммы, то происходит поиск этого двусловия во временной коллекции, содержащей двусловия, определенные ранее за этот же цикл. Если лемма двусловия найдена в коллекции, то значение частоты найденного элемента увеличивается на 1. В противном случае двусловие заносится в коллекцию с частотой 1.

В итоге содержимое коллекции заносится в БД.

2.4 Логика работы программы для классификации по лексико-грамматическим классам

Для отнесения слов и словосочетаний к лексико-грамматическим классам производится обход всех словоупотреблений в БД. Для каждого лексико-грамматического класса производится свой обход с исключением. Пусть лексико-грамматический класс состоит из N слов, тогда при обходе каждых N слов происходит их проверка по маске лексико-грамматического класса. Если какие-либо N слов удовлетворяют маске, то каждое из них помечается как использованное (что исключает вхождение какого-либо из этих слов в другие лексико-грамматические классы), а само словосочетание относится к тому лексико-грамматическому классу, по которому производится обход в данный момент. Обход текста подразумевает обход по каждому из значащих классов с целью уменьшения избыточности выходных данных. Обход производится от классов с большим количеством словоупотреблений к классам с меньшим их количеством.

Следует отметить характер обработки программой входного файла. Одно словоупотребление может быть определено программой Mystem как несколько словоупотреблений, имеющих сходное написание, но разные признаки. В таком случае в БД для этой же позиции (порядковый номер слова в тексте, учитывая различные пунктуационные разделители) заносятся все возможные варианты, определенные программой Mystem. Особое внимание было уделено строгому определению предлогов: если словоупотребление во множестве определенных признаков имеет в том числе предлог, оно больше никак не будет занесено в БД на данной позиции, несмотря на другие варианты, предлагающие включить данное слово как существительное или наречие. Подобная проблема существует и со словами, имеющими варианты как прилагательного, так и существительного (то же самое и с причастиями, определяемыми как глаголы). Здесь видно дальнейшее направление разработки программы – расчет вероятностей всех частей речи для возможности более точного анализа списка терминов-кандидатов.

Во время обхода программа формирует текстовые файлы, содержащие результаты обхода по каждому шаблону.

2.5 Комбинированный метод

Метод выражается в последовательности выполнения операций: подсчет частоты двусловий, отсечение двусловий с частотой 1, фильтрация по лексико-грамматическим классам Сущ. + Сущ., Прил. + Сущ., Прич. + Сущ.

2.6 Логика работы программы для подсчета мер T-score, MI

При нажатии пользователем соответствующей кнопки (запрос подсчета этих мер) происходит загрузка коллекции двусловий из БД. Следует заметить, что до подсчета этих мер пользователь должен инициировать подсчет частот как всех слов в тексте, так и частот двусловий, так как они используются в расчетах этих мер. Далее происходит перебор элементов коллекции двусловий, и для каждого двусловия выполняются следующие действия: по леммам слов в двусловии из таблицы одиночных слов выбираются частоты обоих слов, составляющих двусловие; далее по формулам высчитываются меры и заносятся в соответствующие столбцы перебираемой записи.

*2.7 Логика работы программы для подсчета меры TF*IDF*

После обработки файла, созданного программой Mystem, сразу же запускается метод, высчитывающий

значения TF и IDF для каждого слова: осуществляется перебор слов по частям речи. Для каждого слова заносятся значения TF и IDF в соответствующие столбцы записи, соответствующей рассматриваемому слову в таблице его части речи. Вызов метода расчета этих мер сразу после обработки файла обусловлено используемым алгоритмом обработки файла программой Mystem. После метода обработки файла лемма слова, занесенная в таблицу, содержит также значение частоты слова из Контрастного Корпуса Русского Языка, используемого в выдаче программы Mystem. Вызов метода подсчета TF и IDF устраняет этот недостаток, разделяя лемму и значение из корпуса, высчитывая и занося верные данные в соответствующие столбцы записи.

2.8 Логика работы программы при подсчете мер Log-Likelihood и C-Value

Во многом аналогична логике работы при подсчете мер MI и T-Score. При нажатии соответствующей кнопки происходит перебор коллекции биграмм и подсчет значения с его занесением в соответствующий столбец в таблице биграмм. Как и с упомянутыми в этом пункте методами, подсчету Log-Likelihood и C-Value должны предварять подсчеты частот n-грамм, значения которых используются при вычислениях.

2.9 Оценка качества определения частей речи и классификации

Для отнесения объекта к классу объект должен обладать соответствующим признаком (наличие хотя бы одной запрашиваемой части речи среди определенных программой Mystem частей речи). В результате для каждого регламентного запроса выдается список потенциально устойчивых словосочетаний с разной степенью достоверности вследствие наличия шума, который дает множество определяемых частей речи для одного словоупотребления, а также словосочетания, элементы которых формально имеют признаки, удовлетворяющие лексико-грамматическому классу.

Ранее [8] нами экспериментально были определены значения шума для всех лексико-грамматических классов. Так, для класса Сущ. + Сущ. значение шума составляет 4,37%, для класса Прил. + Сущ. – 0,47%. Однако в данном алгоритме классификации, который оперирует исключительно двусловиями, значение шума для совокупности классов Сущ. + Сущ., Прил. + Сущ. и Прич. + Сущ. составляет 0%. Это объясняется тем, что множественность определений существительных включает только прилагательные, а множественность определений прилагательных – причастия, которые условно отнесены к классу Прил. + Сущ.

3 Эксперимент по автоматизированному извлечению терминологии

Эксперимент по автоматизированному извлечению двухсловных терминов путем комбинирования статистического и лингвистического методов проводится на том же текстовом корпусе по предметной области «Станкостроение», из которого термины были извлечены экспертом.

План эксперимента:

Подготовительные работы

1. Из корпуса экспертом по предметной области «Станкостроение» были выделены все термины. В результате мы получили 373 термина (792 терминоупотребления).

2. Лингвистом был выполнен морфологический разбор словоупотреблений, из которых состоят термины. Так, после упорядочивания по частоте, значимых лексико-грамматических классов было выделено три: Сущ.+Сущ., Прил.+Сущ., Прич.+Сущ.

Проведение эксперимента

1. Выполнить подсчет частоты двусловий, выгрузить результаты.

2. Отсечь двусловия с частотой <=2.

3. К оставшимся двусловиям применить фильтрацию по лексико-грамматическим классам, выгрузить результаты.

Анализ результатов

1. Анализ выдачи программы после подсчета частоты двусловий

Таблица 1

Выдача программы после подсчета частоты двусловий

Частота двусловия	Количество двусловий	Количество терминов	% терминов
2	671	158	23
3	288	81	27
4	122	33	26
5	77	35	45
6	45	21	46
7	33	17	51
8–10	62	35	56
11–61	95	50	52
≥ 2	1393	430	30,86
≥ 5	312	157	50,64

Таким образом, терминологичность даже наиболее частотных двусловий не превышает 56% (см. табл. 1). Среднее значение терминологичности двусловий с частотой >=2 равняется 30,86%. Так, для фильтрации по лексико-грамматическим классам поступит 1393 двусловия.

2. Валидация терминов экспертом по предметной области «Станкостроение»

Выдача программы составила 537 двусловий, из которых эксперт определил терминами 444. Путем качественного сравнения двусловий до и после фильтрации по лексико-грамматическим классам было установлено, что 444 термина после фильтрации идентичны 444 терминам до фильтрации. Таким образом, введением метода фильтрации двусловий с частотой более или равной 2 по лексико-грамматическим классам позволило отсечь 856 двусловий-нетерминов (90% от всех двусловий-нетерминов с частотой более или равной 2) и ни одного двусловия-термина, что выражается в увеличении доли терминов в выдаче при автоматизированном извлечении с 30% до 82%.

3. Сравнение результатов с эталонным списком терминов.

Сравнение с эталонным списком показало, что совпадает 226 терминов, что составляет 42% от выдачи программы или 60% от эталонного списка. Оставшиеся термины, содержащиеся в выдаче программы, присутствуют в эталонном списке как части более многословных терминов.

4. Сравнение результатов с результатами других исследователей.

В работе [9, 10] приводится описание экспериментов по автоматизированному извлечению двухсловных терминов из текста на основании фильтрации по лексико-грамматическим шаблонам и последующего подсчета частоты двусловий (метод Freq) и сравнением с частотой двусловий в контрастном корпусе (метод TF*IDF). В качестве контрастного корпуса используется Web, доступ к которому осуществляется с помощью поисковых машин интернета. Очевидны преимущества такого подхода: по актуальности, разнообразию и размеру с Web не могут конкурировать другие корпусы, а наличие интерфейсов к машинам поиска существенно облегчает реализацию метода.

Вследствие определенных авторами недостатков такого подхода (тематическая несбалансированность Web, различные артефакты – дубликаты документов, опечатки, спам и т. п.) эксперименты проводятся на текстах из трех предметных областей – «Сетевые операционные системы» (СОС), «Философия. Наука. Методология» (ФНМ), «Информационный вестник ВОГиС» (ВОГиС). Результаты подсчета частоты двусловий представлены в двух видах: при строгой и при слабой экспертной оценке.

В нашем случае классификация оценки на строгую и слабую не проводилась. В нашем исследовании принимал участие один эксперт, специалист в анализируемой предметной области. Мы привели сравниваемые результаты к средним значениям, представленным в таблице 2.

Таблица 2

Сравнение результатов

Freq1	TF*IDF	Freq2
67,5%	69%	82%
79%	77,5%	
61,5%	73%	

Выводы:

1. Результаты применения метода Frequency совпадают с результатами других исследователей [9].

2. Результаты последующего применения лингвистического метода показали лучший результат за счет предложенного способа реализации лингвистического метода в виде обхода словосочетаний, начиная с большего.

3. Первый эксперимент по автоматическому извлечению терминологии показал, что комбинирование двух методов позволяет извлекать двухсловные термины из текста с вероятностью 82%.

4. При сравнении с эталонным списком качество извлечения составляет 42% от выдачи программы или 60% от эталонного списка.

Заключение

Разработанная программа классификации двусловий по частоте и лексико-грамматическим шаблонам на основании морфологических признаков, определяемых программой Mystem, должным образом работает по заложенному алгоритму.

Предложенный в работе подход к автоматическому извлечению из текста двухсловных терминов на базе частоты совместной встречаемости словоупотреблений и формализации в виде лексико-грамматических шаблонов лингвистических особенностей употребления терминов на примере предметной области «Станки с ЧПУ» показал лучшие результаты. По результатам проведенного исследования эффективности процедур предложена стратегия объединения результатов их работы, позволяющая улучшить показатели извлечения терминов из отдельно взятого научно-технического текста.

Для повышения вероятности извлечения термина приоритетной задачей становится ввод в алгоритм вероятностно-статистических методов (в первую очередь, C-Value, Mutual Information, Log-likelihood).

СПИСОК ЛИТЕРАТУРЫ

1. Ярушкина Н.Г., Афанасьева Т.В., Перфильева И.Г. Интеллектуальный анализ временных рядов: учеб. пособие. – Ульяновск : УлГТУ, 2010. – 320 с.

2. Интегральный метод принятия решений и анализа нечетких временных рядов /В. Новак [и др.] // Программные продукты и системы.– 2008.– № 4.– С. 18.

3. Афанасьева Т.В., Ярушкина Н.Г. Нечеткое моделирование временных рядов и анализ нечетких тенденций. – Ульяновск : УлГТУ, 2009.

4. Ярушкина Н.Г., Вельмисов А.П., Стецко А.А. Средства data minig для нечетких реляционных серверов данных// Информационные технологии. – 2007.– № 6. – С. 20–29.

5. Афанасьева Т.В, Ярушкина Н.Г. Нечеткий динамический процесс с нечеткими тенденциями в анализе временных рядов // Вестник Ростовского государственного университета путей сообщения. – 2011.– Т. 3. – С. 7–16.

6. Maria Teresa Pazienza1, Marco Pennacchiotti1, and Fabio Massimo Zanzotto Terminology extraction an analysis of linguistic and statistical approaches // Proceedings of the NEMIS 2004 Final Conference, pp. 255–279.

7. Nenadic G., Ananiadou S., McNaught J. Enhancing Automatic Term Recognition through Variation // Proceedings of 20th Int. Conference on Computational Linguistics COLING'04. 2004. pp. 604–610 [Pecina et.al., 2006; Zhang et.al., 2008].

8. Namestnikov A., Yarushkina N. Efficiency of Genetic algorithms for automated design problems // Известия Российской академии наук. Теория и системы управления. – 2002.– № 2.– С. 127–133.

9. Добров Б.В., Лукашевич Н.В., Сыромятников С.В. Формирование базы терминологических словосочетаний по текстам предметной области // Тр. 5-й Всерос. научн. конф. «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» (RCDL-2003). –

СПб., 2003. – С. 201–210.

10. Bourigault, D.: Surface grammatical analysis for the extraction of terminological noun phrases. In: Proc. of Fifteenth International Conference on Computational Linguistics (1992).

11. Daille, B. : Approach mixte pour l'extraction de termilogie: statistique lexicale et filters linguistiques. PhD Thesis, C2V, TALANA, Universitи Paris VII (1994).

12. Patrick Pantel and Dekang Lin, "A Statistical Corpus-Based Term Extractor", Proceedings of 14th Biennial Conference of the Canadian Society on Computational Studies of Intelligence: Advances in Artificial Intelligence, 2001.

13. Загрузить mystem для некоммерческого использования // YANDEX.RU: Яндекс. – URL: http://company.yandex.ru /technologies/mystem/noncommercial.xml (дата обращения 08.11.2013).

14. Андреев И.А., Башаев В.А., Клейн В.В. Разработка программного средства для извлечения терминологии из текста на основании морфологических признаков, определяемых программой Mystem // Интегрированные модели и мягкие вычисления в искусственном интеллекте. – М. : Физматлит. – 2013. с. 1227–1236.

15. Браславский П.И., Соколов Е.А. Сравнение четырех методов автоматического извлечения двухсловных терминов из текста // Компьютерная лингвистика и интеллектуальные технологии. – М. : Изд-во РГГУ. – 2006. – С. 88–91.

16. Браславский П.И., Соколов Е.А. Автоматическое извлечение терминологии с использованием поисковых машин интернета // Компьютерная лингвистика и интеллектуальные технологии. – М. : Изд-во РГГУ. – 2006. – С. 91–94.