УДК 004.522

И.С. Кипяткова

ПРОГРАММНО-АЛГОРИТМИЧЕСКОЕ ОБЕСПЕЧЕНИЕ СОЗДАНИЯ СИНТАКСИЧЕСКО-СТАТИСТИЧЕСКОЙ МОДЕЛИ РУССКОГО ЯЗЫКА ПО ТЕКСТОВОМУ КОРПУСУ

Кипяткова И.С. **Программно-алгоритмическое обеспечение создания синтаксическо-статистической модели русского языка по текстовому корпусу**

Аннотация. Создание модели языка является одним из этапов обучения системы распознавания слитной речи. В статье описаны алгоритм и разработанные программные средства для создания синтаксическо-статистической модели русского языка по текстовому корпусу. Основными этапами в работе алгоритма являются предварительная обработка текстового материала, создание статистической *n*-граммной модели языка, дополнение статистической модели *n*-граммами, полученными в результате синтаксического анализа. Синтаксический анализ позволяет увеличить количество создаваемых в результате обработки текста различных биграмм и тем самым повысить качество модели языка за счет выявления грамматически связанных пар слов. Приводятся результаты тестирования созданных с помощью программного модуля моделей языка по показателям информационной энтропии, коэффициента неопределенности, относительного количества внесловарных слов и совпадений *n*-грамм.

Ключевые слова: автоматическое распознавание речи, статистическая модель языка, синтаксический анализ.

Kipyatkova I.S. **Software for Creation of Sintactico-Statistical Russian Language Model Based on the Text Corpus**

Abstract. Creation of the language model is one of the stages of training of a continuous speech recognition system. In the paper, the developed software for creation of syntactic-statistical Russian language model based on a text corpus is described. The main stages of the algorithm are preliminary text material processing, creation of statistical *n*-gram language model, extension of the statistical model by *n*-grams obtained by syntactical analysis. Syntactical analysis permits to increase the quantity of different bigrams created during text processing and to improve the quality of the language model by extracting grammatically-connected word pairs. The results of the testing of the language models created with the help of the software module are presented.

Keywords: automatic speech recognition, statistical language model, syntactical analysis.

1. Введение. Одним из этапов обучения системы автоматического распознавания слитной речи является создание модели языка. Наиболее распространенным подходом к построению языковых моделей являются статистические *n*-граммы, которые оценивают вероятность появления слова в зависимости от последовательности из *n* произнесенных до этого слов. Существующие модели в большинстве исследований были апробированы на английском языке и, как следствие, не отвечают специфике русского языка, для которого характерны практи-

чески свободный порядок слов в предложениях и наличие большого количества словоформ в каждой лексеме [8, 9, 11, 23, 24].

Для того чтобы учесть дальнодействующие связи между словами во фразе, разработано программно-алгоритмическое обеспечение, позволяющее объединять результаты статистического и синтаксического анализа обучающего текстового корпус и создавать синтаксическо-статистическую модель языка.

2. Анализ существующих подходов статистического анализа естественного языка. Существует несколько вариантов организации моделей языка, основанных на статистическом анализе текста. Для естественного языка чаще всего применяется статистическая модель на основе *n*-грамм, цель которой состоит в оценке вероятности появления цепочки слов $W = (w_1, w_2, ..., w_m)$ в некотором тексте. *n*-граммы представляют собой последовательность из *n* элементов (например, слов), а *n*-граммная модель языка используется для предсказания элемента в последовательности, содержащей *n-1* предшественников [3]. Эта модель основана на предположении, что вероятность какой-то определенной *n*-граммы, содержащейся в неизвестном тексте, можно оценить, зная, как часто она встречается в некотором обучающем тексте.

Вероятность $P(w_1, w_2, ..., w_m)$ можно представить в виде произведения условных вероятностей входящих в нее *n*-грамм [24]:

$$P(w_1, w_2, ..., w_m) = \prod_{i=1}^{m} P(w_i \mid w_1, w_2, ..., w_{i-1})$$

или аппроксимируя $P(w)$ при ограниченном контексте длиной *n-1*:

$$P(w_1, w_2, ..., w_m) \cong \prod_{i=1}^{m} P(w_i \mid w_{i-n+1}, w_{i-n+2}, ...w_{i-1})$$

Вероятность появления *n*-граммы вычисляется на практике следующим образом:

$$P(w_i \mid w_{i-n+1}, ...w_{i-1}) = \frac{C(w_{i-n+1}, ...w_i)}{C(w_{i-n+1}, ...w_{i-1})},$$

где *C* — количество появлений последовательности в обучающем корпусе.

Рассмотрим несколько вариантов статистических моделей, описанных в современной литературе. Модели, основанные на классах (Class-based models), используют функцию, которая отображает каждое слово w_i на класс c_i: f: $w_i \rightarrow f(w_i)=c_i$. Если какой-либо класс содер-

жит более одного слова, то такое отображение приведет к сокращению числа классов по сравнению с количеством слов.

Интервальные модели языка (distance models) помогают включить больший контекст, чем *n*-граммы. В этих моделях биграмма предсказывает слово w_i, основываясь на предшествующем слове w_{i-d}, где d – это расстояние до слова. Другой тип моделей, которые моделируют взаимоотношение пар слов в более длинном контексте, это триггерные модели (Trigger models). Здесь появление инициирующего слова в истории увеличивает вероятность другого слова, называемого целевым, с которым оно связано. Упрощенным вариантом триггерных пар является кэш-модель (cache model). Кэш-модель увеличивает вероятность появления слова в соответствии с тем, как часто данное слово употреблялось в истории, поскольку считается, что, употребив конкретное слово, диктор будет использовать это слово еще раз либо в силу его характерности для конкретной темы, либо потому что диктор имеет тенденцию использовать это слово в своем лексиконе. Кэш-модель можно рассматривать как простую *n*-граммную модель с вероятностями, вычисленными по предшествующей истории слов [28].

Модели, основанные на частях слов, (particle-based models) могут быть эффективными для языков с богатой морфологией (флективных языков). В этом случае слово w разделяется на некоторое число $L(w)$ частей (морфем) с помощью функции U: $w \rightarrow U(w)=u^1, u^2, \ldots, u^{L(w)}$, $u^i \in \Psi$, где Ψ - это набор частей слова. Разделение слов на морфемы можно производить двумя путями: при помощи словарных и алгоритмических методов [21]. Преимуществом алгоритмических методов является то, что они опираются лишь на анализ текста и не используют никаких дополнительных знаний, что позволяет анализировать текст на любом языке. Преимуществом словарных методов является то, что они позволяют получить правильное разбиение слов на морфемы, а не на псевдоморфемные единицы (как в алгоритмических методах), что может быть использовано далее на уровне пост-обработки гипотез распознавания фраз.

n-граммы переменной длины (varigrams) не ограничивают последовательности слов до определенного *n*, а вместо этого хранят различные последовательности разной длины [23]. По существу, они могут рассматриваться как *n*-граммные модели с большим *n* и такими принципами сокращения длины моделей, которые сохраняют только небольшой поднабор всех длинных последовательностей, встретившихся в обучающем тексте.

В работе [12] предлагается дальнодействующая триграммная модель, в которой разрешены связи между словами, находящимися не только в пределах двух предыдущих слов, но и на большем расстоянии от предсказываемого слова. Лежащий в основе модели грамматический анализатор позволяет связать между собой пары зависимых слов, которые могут отстоять друг от друга на несколько разделяющих слов.

В работе [17] предлагаются составные языковые модели с введением понятия категорной языковой модели и, в частности, категорных *n*-грамм. Каждому слову в словаре приписываются 15 атрибутов, определяющих грамматические свойства словоформы. Множество значений атрибутов определяет класс словоформы. Каждое слово в предложении рассматривается как его начальная форма и морфологический класс. В итоге языковая модель разбивается на две составляющие: изменяемую часть (основанную на морфологии) и постоянную часть (основанную на начальных формах слов), которая строится как *n*-граммная языковая модель.

В работе [26] представлена стохастическая морфо-синтаксическая модель венгерского языка для системы распознавания речи. Эта модель описывает допустимые в венгерском языке словоформы (комбинации морфем). Применение морфо-синтаксической модели языка позволило уменьшить количество неправильно распознанных морфем на 17,9% по сравнению с базовой триграммной моделью.

В работе [25] предложена концепция синтаксических *n*-грамм (*sn*-грамм). В случае *sn*-грамм соседние слова выбираются в соответствии с их синтаксическими связями в синтаксических деревьях, а не в соответствии с тем, как они появились в тексте. В работе используются традиционные *n*-граммы слов, а также метки и характеристики частей речи. В данной статье *sn*-граммы применяются только для задачи определения авторства, но также предлагается использовать *sn*-граммы для задач распознавания и перевода.

В [19] описывается структурированная языковая модель, которая применяется на стадии декодирования речи для синтаксического разбора результатов распознавания, синтаксическое дерево строится динамически в ходе распознавания. Модель позволяет использовать дальнодействующие связи между словами и предсказывать слово не только по нескольким предыдущим лексическим единицам, но также по доступным главным словам. В статье сообщается об уменьшении ошибок распознавания английской речи при применении структурированной модели по отношению к распознаванию с использованием триграммной модели. Такая модель хорошо подходит для аналитических

языков с жесткой грамматической структурой (например, для английского языка), однако для языков с более свободной грамматикой приемлемым является только разбор предложения целиком, потому что грамматически связанные слова могут быть расположены в предложении достаточно далеко друг от друга.

В данной работе статистическая биграммная модель языка расширена за счет синтаксического анализа обучающего текстового корпуса. В ходе статистического анализа из-за нежесткого порядка слов в русском языке не создаются биграммы, содержащие грамматически связанные пары слов, которые были разделены в обучающем тексте другими словами [7]. Синтаксический анализ позволяет увеличить количество создаваемых в результате обработки текста различных биграмм и тем самым повысить качество модели языка за счет выявления грамматически связанных пар слов.

Существует несколько синтаксических анализаторов для русского языка, например, один разработан в компании ООО «Диктум» [14], второй входит в систему морфо-синтаксического анализа «Treeton» [16], третий входит в лингвистический процессор ЭТАП-3 [4], четвертый — RCO Syntactic Engine SD [2], пятый (для русского и английского языков) — входит в пакет программ Cognitive Dwarf компании Cognitive Technologies [1], шестой описан в работе [6], седьмой — проект АОТ [15]. Для использования в данной работе был выбран последний анализатор, имеющий ряд преимуществ перед аналогами, в частности, он распространяется с открытым исходным кодом и базами данных, допускающими модификацию, обеспечивает высокую скорость обработки текстовых данных, также в своей основе он использует постоянно обновляемую грамматическую базу данных, основанную на базовом грамматическом словаре А.А. Зализняка [5] и расширенную разработчиками.

3. Разработка средств синтаксическо-статистического моделирования русского языка. В данном разделе представлено описание разработанного программно-алгоритмического обеспечения создания синтаксическо-статистической модели русского языка по текстовому корпусу. В ходе анализа текстового материала производится разделение текста на отдельные предложения, расшифровка общепринятых сокращений, удаление знаков препинания, затем выполняется создание статистической *n*-граммной модели языка, дополнение статистической модели *n*-граммами, полученными в результате синтаксического анализа. В ходе синтаксического анализа обучающего текстового корпуса выявляются грамматически связанные пары слов, которые были разде-

лены в обучающем тексте другими словами. Синтаксический анализ позволяет увеличить количество создаваемых в результате обработки текста различных биграмм и тем самым повысить качество модели языка за счет выявления грамматически связанных пар слов. Схема алгоритмической модели, применяемой при обработке текстового материала для создания модели языка, представлена на рисунке 1.

Программная реализация алгоритма создания синтаксическо-статистической модели языка была выполнена на языке программирования Perl, с использование внешних модулей в виде исполняемых файлов, в том числе модули комплексов программ The CMU-Cambridge Statistical Language Modeling Toolkit (CMU SLM) [20], АОТ (Автоматическая обработка текста) [15].

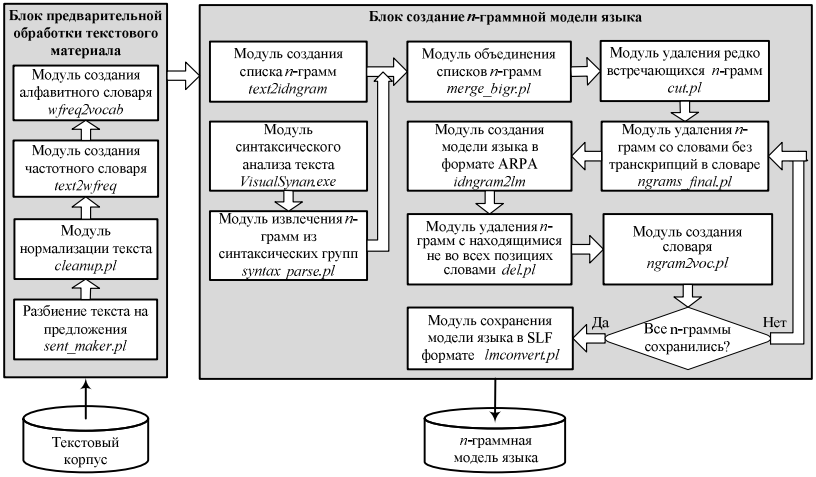

Рис. 1. Алгоритм создания синтаксическо-статистической модели языка.

В блоке предварительной обработки текстового материала осуществляется фильтрация и нормализация обучающего текстового корпуса, который будет в дальнейшем использоваться для создания модели языка, на основе следующего скрипта:

perl sent_maker.pl text.txt text_out.txt
perl cleanup.pl text_out.txt text_clean.txt
text2wfreq <text_clean.txt> dict.wfreq
wfreq2vocab -top A <dict.wfreq> dict.vocab

Для запуска скриптов на языке Perl предварительно на компьютер следует установить транслятор с этого языка, например Active Perl или аналогичный. Модуль *sent_maker.pl* осуществляет разбивку текста на предложения, удаление текста, написанного в любых скобках, а также предложений, состоящих из пяти и меньшего количества слов (как правило — это заголовки, составленные не по грамматическим правилам для полных предложений). На вход модуля поступает файл *text.txt* с текстом для обучения модели языка, на выходе создается файл *text_out.txt*, содержащий разбитый на предложения текст.

После этого производится нормализация текстового корпуса с помощью модуля *cleanup.pl*. Начало и конец предложения отмечаются метками <s> и </s> соответственно. Затем из текстов удаляются знаки препинания, символы "№" и "#" меняются на слово "номер". Все числа и цифры объединяются в единый класс, который обозначается в результирующем тексте символом "№". За одно число принимается группа цифр, которые могут быть разделены точкой, запятой, пробелом или тире. Также символом "№" обозначаются римские цифры, представляющие собой совокупность латинских букв *I*, *V*, *X*, *L*, *C*, *D*, *M*, которые могут быть разделены пробелом или тире. В отдельные классы выделяются интернет-адреса (обозначаются знаком "<>") и E-mail адреса (обозначаются символом "<@>"). Затем происходит расшифровка общепринятых сокращений (например, «см», «кг»). В словах, начинающихся с заглавной буквы, происходит замена заглавной буквы на строчную. Если все слово написано заглавными буквами, то замена не делается, так как это слово, вероятно, является аббревиатурой. Знаки препинания из текстов удаляются. На выходе модуля создается текстовый файл *text_clean.txt*, содержащий обработанный и нормализованный текстовый корпус.

Модуль создания частотного словаря *text2wfreq.exe* является исполняемым модулем CMU SML [20]. На вход модуля поступает текст из файла *text_clean.txt*, а на выходе создается частотный словарь *dict.wfreq*, который представляет собой список всех словоформ, используемых в тексте, с частотой их встречаемости в тексте.

Следующий модуль *wfreq2vocab.exe* из комплекса программ CMU SLM создает из частотного словаря алфавитный словарь *dict.vocab* словоформ, используемых в тексте. Размер словаря можно задать двумя способами: напрямую указав максимальный размер словаря, либо указав максимальный порог частоты появления слова, начиная с которой слова будут записываться в словарь. Параметр *-top* задает максимальный размер словаря, то есть в словарь будет записано *A* наиболее

частотных слов. Также может быть использован параметр *-gt*, устанавливающий порог *B* для значения частоты появления слова, начиная с которой слова будут записываться в словарь. Если параметры *-top* и *-gt* не указаны, то в словарь будет записано 20000 слов с наибольшей частотой.

В результате работы блока предварительной обработки текстового материала создается нормализованный обучающий текстовый корпус, а также частотный и алфавитный словари словоформ. Далее в блоке создания модели языка производится ряд операций, обеспечивающих статистическую обработку текстового корпуса, создание списка *n*-грамм, а также статистической и синтаксической моделей языка с их последующим объединением. Работа блока осуществляется на основе следующего скрипта:

text2idngram -vocab dict.vocab -n N -write_ascii <text_clean.txt> idngram.txt

VisualSynan.exe

perl syntax_parse.pl syntax.txt id_bigr_syntax.txt

perl merge_bigr.pl id_bigr_syntax.txt idngram.txt bigr_syn_stat.txt

perl cut.pl bigr_syn_stat.txt dict.vocab bigr_cutoff.txt K

perl ngrams_final.pl bigr_cutoff.txt transcr.txt dict.vocab voc_new.txt ngrams_final.txt

idngram2lm.exe -idngram ngrams_final.txt -vocab voc_new.txt -arpa lm.arpa -ascii_input -n 2 -calc_mem -vocab_type T

perl del.pl lm.arpa ngrams_final.txt voc_new.txt transcr.txt ngrams_final_del.txt 2

perl ngram2voc.pl ngrams_final_del.txt voc_new.txt voc_new_del.txt

perl ngrams_final.pl ngrams_final_del.txt voc_new_del.txt voc_new.txt words.txt ngrams_final(final).txt

idngram2lm.exe -idngram ngrams_final(final).txt -vocab words.txt -arpa lm(final).arpa -ascii_input -n 2 -calc_mem -vocab_type T

perl lmconvert.pl lm(final).arpa words.txt lm(final).slf 2

Первым запускается модуль *text2idngram.exe* из состава CMU SLM, который создает файл *idngram.txt* со списком *n*-грамм и их частотой появления в обучающем корпусе. Индекс слова в *n*-грамме соответствует порядковому номеру слова в алфавитном словаре. При запуске программы необходимо указать *N* – значение *n* для *n*-грамм.

Затем используется открытый парсер VisualSynan проекта AOT, на выходе которого формируется список синтаксических групп слов в предложениях. После чего модуль *syntax_parse.pl* из результатов синтаксического анализа, сохраненных в файле *syntax.txt*, выбирает только

те синтаксические группы, которые были в тексте разделены другими словами, то есть дальнодействующие связи. В результате создается файл *id_bigr_syntax.txt*, содержащий список синтаксических групп, которые в тексте были разделены другими словами, с частотой их появления в обучающем корпусе.

Следующий модуль *merge_bigr.pl* объединяет списки биграмм, созданные в результате статистической обработки биграмм и синтаксического анализа текста, и сохраняет их в файл *bigr_syn_stat.txt*.

Для увеличения скорости обработки целесообразно удалять редкие *n*-граммы. Модуль *cut.pl* удаляет *n*-граммы, частота появления которых меньше заданного порога *K*, и создает файл *bigr_cutoff.txt*.

Далее модуль *ngram_final.pl* удаляет *n*-граммы со словами, отсутствующими в словаре транскрипций. При запуске программы необходимо указать файл *transcr.txt* со словарем транскрипций для слов из собранного корпуса. В результате работы модуля создается словарь *voc_new.txt*, содержащий только те слова, транскрипции для которых присутствовали в файле *transcr.txt*, а также файл *ngrams_final.txt* со списком *n*-грамм, в этом файле индексы *n*-грамм соответствует словарю *voc_new.txt*.

Модуль *idngram2lm* из состава CMU SLM создает вероятностную модель языка в формате ARPA. При запуске модуля указываются следующие параметры: *lm.arpa* – выходной файл, содержащий модель языка в формате ARPA, *calc_mem* – допустимый объем оперативной памяти, *vocab_type T* – тип словаря. При T=0 словарь является закрытым, это означает, что в модели языка нет внесловарных слов, то есть слов, отсутствующих в словаре *voc_new.txt*. Открытый словарь разрешает появление внесловарных (неизвестных для словаря) слов, при этом все внесловарные слова будут обозначены одним символом "<UNK>". Открытый словарь может быть двух видов: T=1 означает, что внесловарные слова будут трактоваться так же, как и любые другие слова в словаре; T=2 предполагает, что в модели языка нет внесловарных слов, но они могут быть в текстовых данных, которые будут использоваться для тестирования модели языка.

Фрагмент созданной модели языка в формате ARPA показан на рисунке 2а. В начале идет список униграмм, слева от униграммы указывается значение десятичного логарифма вероятности ее появления, справа – коэффициент возврата (back-off weight) [22], который применяется в тех случаях, когда некоторая *n*-грамма отсутствует в обучающем корпусе или частота ее появления очень низкая, тогда вместо нее

используется вероятность (n-1)-граммы, умноженная на коэффициент возврата. Ниже идет список биграмм с вероятностями их появления.

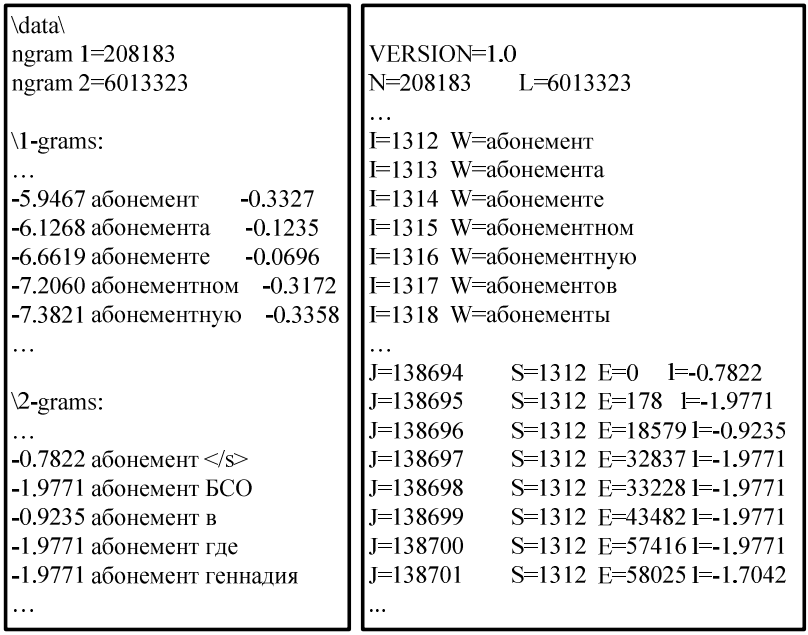

Рис. 2. Фрагменты моделей языка в формате: а) ARPA; б) SLF.

Для удаления n-грамм со словами, присутствующими не во всех позициях, применяется модуль del.pl, в результате создается выходной файл *ngrams_final_del.txt*. Затем модуль *ngram2voc.pl* создает словарь и сохраняет его в файл *voc_new_del.txt* по получившемуся списку биграмм и исходному алфавитному словарю.

Если были удалены n-граммы, то с учетом вновь созданного словаря далее необходимо снова запустить программу *ngrams_final.pl*, где вместо словаря транскрипций указать словарь, созданный программой *ngram2voc.pl*. В полученном списке биграмм индексы слов будут соответствовать словарю *words.txt*. После удаления n-грамм создается новая модель языка в файле *ngrams_final(final).txt* с помощью программного модуля *idngram2lm*.

Модуль *lmconvert.pl* перевода модели языка в формат SLF (Standard Lattice Format) трансформирует модель языка из формата ARPA в более компактный формат, применяемый в комплексе HTK (см. рисунок 2б).

Описанные программные средства предназначены для предварительной обработки текстового материала, а также для создания статистической *n*-граммной или синтаксическо-статистической моделей русского языка, применяемых в различных информационно-управляющих системах с речевыми и многомодальными интерфейсами [10, 13].

4. Экспериментальное исследование языковых моделей. Созданные в ходе тестирования модели языка оценивались по показателям информационной энтропии, коэффициента неопределенности и количества внесловарных слов. По определению [18] информационная энтропия – мера хаотичности информации, неопределенность появления какого-либо символа первичного алфавита. При отсутствии информационных потерь она численно равна количеству информации на символ передаваемого сообщения. Поскольку тексты на естественном языке могут рассматриваться в качестве информационного источника, энтропия вычисляется по следующей формуле [23]:

$$H = -\lim_{m\to\infty}\frac{1}{m}\sum_{w_1,w_2,\ldots,w_m}\left(P(w_1,w_2,\ldots,w_m)\log_2 P(w_1,w_2,\ldots,w_m)\right)$$

Это суммирование делается по всем возможным последовательностям слов текстового корпуса. Но поскольку язык является эргодичным источником информации (случайный процесс эргодичен [18], если все его статистические характеристики с вероятностью, сколь угодно близкой к единице, можно предсказать по одной реакции из ансамбля с помощью усреднения по времени) [23], выражение для вычисления энтропии будет выглядеть следующим образом:

$$\hat{H} = -\frac{1}{m}\log_2 P(w_1,w_2,\ldots,w_m)$$

Коэффициент неопределенности (perplexity) является параметром, по которому оценивается сложность *n*-граммных моделей языка, вычисляется следующим образом [23]:

$$PP = 2^{\hat{H}} = \hat{P}(w_1,w_2,\ldots,w_m)^{-\frac{1}{m}},$$

где $\hat{P}(w_1,w_2,\ldots,w_m)$ – вероятность последовательности слов $w_1, w_2,\ldots,w_m$. С практической точки зрения, коэффициент неопреде-

ленности показывает, сколько в среднем различных наиболее вероятных слов может следовать за данным словом.

При создании модели языка был собран и обработан текстовый русскоязычный корпус, сформированный из новостных лент последних лет четырех интернет-сайтов: www.ng.ru («Независимая газета»), www.smi.ru («СМИ.ru»), www.lenta.ru («LENTA.ru»), www.gazeta.ru («Газета.ru»). Обучающий текстовый материал собирается с интернет-сайтов электронных газет, поскольку в них присутствуют материалы интервью и выступлений, содержащие предложения со спонтанной речью. Для создания модели языка, использующейся в системе распознавания речи, такие текстовые материалы особенно важны, так как позволяют учесть вариативность, характерную для разговорной речи. Общий объем собранного корпуса после его обработки и нормализации составляет свыше 110 млн. словоупотреблений (около 750 Мб данных). Для оценки модели языка использовались 100 фраз, взятых из материалов интернет-газеты: www.fontanka.ru («Фонтанка.ru»).

В таблице 1 представлены результаты вычисления энтропии и коэффициента неопределенности для различных моделей языка.

Таблица 1. **Параметры созданных моделей языка**

Тип модели языка	Размер словаря, тыс. слов	Энтропия, бит/слово	Коэффициент неопределенности
1-граммная	79	12,1	4499
1-граммная	208	12,4	5493
1-граммная	535	12,5	5851
1-граммная	2350	12,5	5909
2-граммная	79	9,7	851
2-граммная	208	9,6	777
2-граммная с синтаксическим анализом	210	9,6	772
3-граммная	76	8,8	452
3-граммная	235	9,2	594

Для тестовых фраз было определено относительное количество внесловарных слов и процент совпадений *n*-грамм – количество *n*-грамм в тестовых данных, которые присутствуют в модели языка. Результаты представлены в таблице 2.

Таблица 2. **Относительное количество внесловарных слов и совпадений *n*-грамм для тестового корпуса при использовании различных моделей языка**

Тип модели языка	Размер словаря, тыс. слов	Относительное количество внесловарных слов, %	Относительное количество совпадений *n*-грамм, %
1-граммная	79	3,5	96,5
1-граммная	208	0,8	99,3
1-граммная	535	0,1	99,9
1-граммная	2350	0,0	100,0
2-граммная	79	3,5	72,6
2-граммная	208	0,8	83,6
2-граммная с синтаксическим анализом	210	0,8	84,2
3-граммная	76	4,9	35,8
3-граммная	235	0,7	34,7

Полученные значения коэффициента неопределенности и относительного количества внесловарных слов являются достаточно большими. Например, для английского языка при размере словаря в 200 тыс. слов, коэффициент неопределенности для биграмм равен 232 [23], при этом энтропия приблизительно равна 7,9 бит/слово, а относительное количество внесловарных слов составляет 0,31 % для тестового корпуса объемом 1,12 млн слов.

4. Заключение. В статье представлено программно-алгоритмическое обеспечение для создания синтаксическо-статистической модели языка. Данная модель языка создается на основе базовой биграммной модели, полученной в результате автоматического статистического анализа обучающего текстового корпуса, собранного с интернет-сайтов электронных газет. Затем базовая биграммная модель расширяется за счет выполнения синтаксического анализа обучающего текстового корпуса, в ходе которого выявляются грамматически связанные пары слов, разделенные в тексте другими словами. Таким образом, синтаксический анализ позволяет учесть дальнодействующие грамматические связи между словами. В зависимости от задачи в качестве обучающего текстового корпуса могут использоваться тексты из различных предметных областей, чтобы настроить предметно-ориентированную систему автоматического распознавания речи.

Литература

1. *Антонова А.А., Мисюрев А.В.* Об использовании синтаксического анализатора Cognitive Dwarf 2.0 // Труды ИСА РАН. Т 38, 2008. С 91-109.
2. Библиотека синтаксического анализа текста RCO Syntactic Engine SDK. http://www.rco.ru/product.asp?ob_no=15 (дата обращения: 15.01.2013).
3. *Джелинек Ф.* Распознавание непрерывной речи статистическими методами // ТИИЭР, 1976. Т. 64. № 4. С. 131-160.
4. *Дружкин К.Ю., Цинман Л.Л.* Синтаксический анализатор лингвистического процессора ЭТАП-3: Эксперименты по ранжированию // Материалы международной конференции «Диалог 2008». Москва, 2008.
5. *Зализняк А.А.* Грамматический словарь русского языка: Словоизменение // 4-е изд., испр. и доп. М.: Руские словари, 2003.
6. *Кагиров И.А., Леонтьева Ан.Б.* Автоматический синтаксический анализ русских текстов на основе грамматики составляющих // Известия вузов. Приборостроение, 2008. Т. 51, № 11. С. 47-51.
7. *Кипяткова И.С.* Комплекс программных средств обработки и распознавания разговорной русской речи // Информационно-управляющие системы, 2011. № 4(53), С. 53-59.
8. *Кипяткова И.С., Карпов А.А.* Разработка и исследование статистической модели русского языка // Труды СПИИРАН. СПб: СПИИРАН, 2010. №1(12). С. 35-49.
9. *Кипяткова И.С., Карпов А.А.* Автоматическая обработка и статистический анализ новостного текстового корпуса для модели языка системы распознавания русской речи // Информационно-управляющие системы. СПб: СПбГУАП, 2010. № 4(47). С. 2-8.
10. *Кипяткова И.С., Карпов А.А.* Эксперименты по распознаванию слитной русской речи с использованием сверхбольшого словаря // Труды СПИИРАН. СПб: СПИИРАН, 2010. №1(12). С. 63-74.
11. *Леонтьева Ал.Б., Кипяткова И.С.* Учет особенностей спонтанной речи при создании систем автоматического распознавания // Известия вузов. Приборостроение, 2008. Т. 51. № 11. С. 51-56.
12. *Протасов С.В.* Вывод и оценка параметров дальнодействующей триграммной модели языка // Материалы международной конференции «Диалог 2008». Москва, 2008. С. 443-449.
13. *Ронжин Ал.Л., Ронжин Ан.Л.* Система аудиовизуального мониторинга участников совещания в интеллектуальном зале // Доклады ТУСУРа, 2011. № 1 (22), часть 1. С. 153-157.
14. Сайт компании «ООО Диктум». http://www.dictum.ru/ (дата обращения: 15.01.2013).
15. *Сокирко А.В.* Морфологические модули на сайте www.aot.ru // Труды Международной конференции «Диалог-2004. Компьютерная лингвистика и интеллектуальные технологии». М.: Наука, 2004. С. 559-564.
16. *Старостин А.С., Мальковский М.Г.* Алгоритм синтаксического анализа, используемый в системе морфо-синтаксического анализа «Treeton» // Материалы международной конференции «Диалог 2007». Москва, 2007. С. 516-524.
17. *Холоденко А.Б.* О построении статистических языковых моделей для систем распознавания русской речи // Интеллектуальные системы, 2002. Т.6. Вып. 1-4. С. 381-394.
18. *Шеннон К.* Работы по теории информации и кибернетике. М.: Изд. иностр. лит., 2002.

19. *Chelba C., Jelinek F.* Structured language model // Computer Speech and Language, 2000. Vol. 10. pp. 283-332.
20. *Clarkson P., Rosenfeld R.* Statistical language modeling using the CMU-Cambridge toolkit // Proc. of EUROSPEECH. Rhodes. Greece, 1997. pp. 2707–2710.
21. *Kurimo M., Hirsimäki T., Turunen V.T., Virpioja S., Raatikainen N.* Unsupervised decomposition of words for speech recognition and retrieval // In Proceedings of 13-th International Conference «Speech and Computer» SPECOM'2009. St. Petersburg, 2009. pp. 23-28.
22. *16. Merkel A., Klakow D.* Improved Methods for Language Model Based Question Classification // Proceedings of 8th Interspeech Conference. Antwerp, 2007. pp. 322-325.
23. *Moore G.L.* Adaptive Statistical Class-based Language Modelling // PhD thesis. Cambridge University, 2001. 193 p.
24. *Rabiner L., Juang B.H.* Fundamentals of Speech Recognition. Prentice Hall. 1993. 507 p.
25. *Sidorov G., Velasquez F., Stamatatos E., Gelbukh A., Chanona-Hernández L.* Syntactic Dependency-based N-grams as Classification Features, Springer LNAI 7630, Mexico, 2012. pp. 1-11.
26. *Szarvas M., Furui S.* Finite-state transducer based modeling of morphosyntax with applications to Hungarian LVCSR // Proc. ICASSP'2003, Hong Kong, China, 2003. pp. 368–371.
27. The CMU Statistical Language Modeling (SLM) Toolkit. http://www.speech.cs.cmu.edu/SLM_info.html (дата обращения: 15.10.2012).
28. *Vaičiūnas A.* Statistical Language Models of Lithuanian and Their Application to Very Large Vocabulary Speech Recognition // Summary of Doctoral Dissertation. Vytautas Magnus University. Kaunas, 2006. 35 p.
29. *Whittaker E.W.D.* Statistical Language Modelling for Automatic Speech Recognition of Russian and English // PhD thesis. Cambridge University, 2000. 140 p.

Кипяткова Ирина Сергеевна — канд. техн. наук**,** старший научный сотрудник лаборатории речевых и многомодальных интерфейсов СПИИРАН. Область научных интересов: автоматическое распознавание речи, статистические модели языка. Число научных публикаций — 40. kipyatkova@iias.spb.su; СПИИРАН, 14-я линия В.О., д. 39, г. Санкт-Петербург, 199178, РФ; р.т. +7(812)328-7081, факс +7(812)328-7081.

Kipyatkova Irina Sergeevna — PhD, senior researcher, Laboratory of Speech and Multimodal Interfaces, SPIIRAS. Research interests: automatic speech recognition, statistical language models. The number of publications — 40. kipyatkova@iias.spb.su; SPIIRAS, 39, 14-th Line V.O., St. Petersburg, 199178, Russia; office phone +7(812)328-7081, fax +7(812)328-7081.

Поддержка исследований. Данное исследование поддержано Минобрнауки РФ (ФЦП «Исследования и разработки», госконтракт № 07.514.11.4139) и фонда РФФИ (проект № 12-08-01265-а).

Рекомендовано лабораторией речевых и многомодальных интерфейсов, заведующий лабораторией Ронжин А.Л, д-р техн. наук, доц.
Статья поступила в редакцию 31.01.2013.

РЕФЕРАТ

Кипяткова И.С. **Программно-алгоритмическое обеспечение создания синтаксическо-статистической модели русского языка по текстовому корпусу.**

Создание модели языка является одним из этапов обучения системы распознавания слитной речи. Наиболее распространенные статистические модели на основе *n*-грамм слов являются недостаточно эффективными для русского языка. Ввиду нежесткого порядка слов в русском языке многие грамматически связанные пары слов оказываются в предложении разделены другими словами, и в результате статистического анализа текста не создаются биграммы, содержащие такие пары слов. Увеличить количество создаваемых в результате обработки текста различных *n*-грамм и тем самым повысить качество модели языка позволяет выявление грамматически связанных пар слов за счет синтаксического анализа предложений обучающего корпуса.

В статье описывается программно-алгоритмическое обеспечение для создания синтаксическо-статистической модели языка. В ходе анализа текстового материала производится разделение текста на отдельные предложения, расшифровка общепринятых сокращений, удаление знаков препинания, затем выполняется создание статистической *n*-граммной модели языка, дополнение статистической модели *n*-граммами, полученными в результате синтаксического анализа. В состав программно-алгоритмического обеспечения входят блок предварительной обработки текстового материала и блок создания *n*-граммной модели языка. В блоке предварительной обработки текстового материала осуществляется фильтрация и нормализация обучающего текстового корпуса, который будет в дальнейшем использоваться для создания модели языка. В блоке создания модели языка производится ряд операций, обеспечивающих статистическую обработку текстового корпуса, создание списка *n*-грамм, а также статистической и синтаксической моделей языка с их последующим объединением.

При создании модели языка был собран и обработан текстовый русскоязычный корпус, сформированный из новостных лент последних лет четырех интернет-сайтов. Были созданы униграммная, биграммная и триграммная модели с различным размером словаря, а также ситнаксическо-статистическая модель со словарем 210 тыс. слов. Созданные в ходе тестирования программного модуля модели языка оценивались по показателям информационной энтропии, коэффициента неопределенности, относительного количества внесловарных слов и совпадений *n*-грамм. Для синтаксическо-статистической модели языка коэффициент неопределенности был равен 772, энтропия – 9,6 бит/слово, относительное количество внесловарных слов – 0,8, относительное количество совпадений *n*-грамм – 84,2.

SUMMARY

Kipyatkova I.S. **Software for Creation of Sintactico-Statistical Russian Language Model Based on the Text Corpus.**

Creation of the language model is one of the stages of the training of the continuous speech recognition system. The most widely used statistical models based on *n*-grams are inefficient for Russian language. Because of the free word order in Russian language many grammatically-connected word pairs are separated by other words in a sentence, and bigrams containing such word pairs are not created during the statistical analysis. Extracting grammatically-connected by syntactical analysis of the sentences of the training corpus allows to increase the quantity of different bigrams created during text processing and to improve the quality of the language model.

In the paper, a software for creation of sintactico-statistical Russian language model is described. The developed software carries out the preliminary processing of the text material (dividing text into sentences, interpretation of the conventional abbreviation, and deletion of punctuation marks), creation of statistical *n*-gram language model, and extension of the statistical model by *n*-grams obtained by syntactical analysis. The software includes the block of preliminary text material processing and the block of *n*-gram language model creation. In the block of preliminary text material processing filtration and normalization of the training text corpus, which will be further used for language model creation, is performed. In the block of the language model creation a sequence of operations providing the statistical processing of the text corpus, creation of the *n*-gram list, statistical, syntactical and combined language model is carried out.

The Russian text corpus assembled from recent news of four Internet sites was collected and processed. Unigram, bigram, and trigram language models with different size of vocabulary as well as syntactic-statistical model with 210K vocabulary were created. Language models created during software module testing were estimated using information entropy, perplexity, out-of-vocabulary rate, and *n*-gram hit. For the syntactic-statistical language model perplexity was 772, entropy equals 9.6 bit/word, out-of-vocabulary rate was 0.8, and hit rate was 84.2.