УДК 004.912: 811.512.111

РЕЗУЛЬТАТЫ ТЕСТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ НАЦИОНАЛЬНОГО КОРПУСА ЧУВАШСКОГО ЯЗЫКА

Желтов В.П., Желтов П.В., Сергеев Е.С.

ФГБОУ ВО «Чувашский государственный университет имени И.Н. Ульянова», Чебоксары, e-mail: akaevgeniy@rambler.ru

В статье рассматриваются вопросы тестирования и отладки программного комплекса национального корпуса чувашского языка. Приводятся результаты тестирования морфологического и синтаксического анализатора чувашского языка. Тестирование производилось по двум основным направлениям: проверка корректности работы системы и быстродействие программного обеспечения. При тестировании морфологического анализатора в качестве входной информации использовались слова с несколькими значениями, слова-исключения, слова с максимальным количеством букв и т.д. Синтаксический анализатор протестирован на предмет нахождения некорректного разбора простых и сложных предложений. Во всех случаях программа производила точный анализ с правильным определением всех характеристик слова с учетом части речи, типов предложения, основ предложения и др. показателей. Основные результаты проиллюстрированы в виде экранных форм и сведены в таблицу тестирования. Быстродействие системы проверялось при помощи встроенного класса Stopwatch среды Visual Studio. Слова с максимальным количеством букв обрабатываются не более чем за 3 миллисекунды. Процесс анализа простого и сложного предложения при помощи настоящего программного обеспечения по продолжительности практически не отличается, завершается за 3–4 миллисекунды.

Ключевые слова: тестирование, морфологический анализ, компьютерная лингвистика, лингвопроцессор, чувашский язык, синтаксический анализ, лингвистический корпус

RESULTS OF TESTING THE SOFTWARE OF THE NATIONAL CORPORA OF THE CHUVASH LANGUAGE

Zheltov V.P., Zheltov P.V., Sergeev E.S.

Federal State Budget Educational Institution of Higher Education «Chuvash State University named after I.N. Ulyanov», Cheboksary, e-mail: akaevgeniy@rambler.ru

The article discusses the testing and debugging of the software package of the national hull of the Chuvash language. The results of testing the morphological and syntactic analyzer of the Chuvash language are presented. The testing was carried out in two main directions: checking the correctness of the system and the speed of the software. When testing the morphological analyzer, words with several meanings, words-exceptions, long words, etc., were used as input information. The parser has been tested to find an incorrect analysis of simple and complex sentences. In all cases, the program produced an accurate analysis with the correct definition of all the characteristics of the word, taking into account the part of speech, the types of supply, the fundamentals of the proposal, and other indicators. The main results are illustrated in the form of screen forms and are summarized in the test table. The speed of the system was checked using the built-in Stopwatch class of Visual Studio. The longest words are processed no more than 3 milliseconds. The process of analyzing a simple and complex sentence with the help of this software for the duration is practically the same, ending in 3–4 milliseconds.

Keywords: testing, morphological analysis, computer linguistics, linguoprocessor, Chuvash language, syntactic analysis, linguistic corpus

Морфологический и синтаксический анализ естественного языка, как важные составляющие компьютерных лингвистических систем, необходимы для реализации механизма поиска, обслуживания национального корпуса языка, представления в качестве выходных данных морфологических и синтаксических составляющих текста (входной информации) [1–3].

В настоящей статье рассматриваются вопросы тестирования разработанного программного обеспечения национального корпуса чувашского языка, представляющего собой лингвопроцессор. На этапе проектирования морфологического и синтаксического анализатора выявлены характерные особенности чувашского языка, обработку которых необходимо реализовать в программном обеспечении (ПО) для корректности работы ПО [4]. Исходными целями разработки лингвопроцессора являются реализация морфологического анализа слов (соответствие лексико-грамматического класса), синтаксического анализа простых и сложных предложений чувашского языка [5, 6]. Морфологический анализатор национального корпуса чувашского языка реализован на основе теории, изложенной в [7, с. 83–87; 8, с. 120–123, и включает в себя следующие компоненты:

1. dll-библиотеку, содержащую морфологический анализатор и реализованную на языке C# с использованием платформы .NET Microsoft.

2. Вкладку «Морфологический анализатор» на сайте национального корпуса чувашского языка (http://yuman21.ru/Morf).

Вкладка «Морфологический анализатор» содержит следующие элементы пользовательского интерфейса:

1) текстовое поле для ввода анализируемого слова на чувашском языке с кнопками для ввода специальных чувашских символов (ă, ĕ, ӳ, ç);

2) кнопку «Анализ», расположенную рядом с текстовым полем.

Морфологический анализатор, реализованный офлайн, содержит окно отображения результатов морфологического анализа как для отдельно введенного слова или фразы, так и возможность вставлять текст для анализа в специальном поле текствого ввода.

База данных морфологического анализатора состоит из словаря основ чувашского языка и базы аффиксов.

Словарь представляет собой текстовый файл, в котором слова представлены следующим образом: слово, часть речи, информация об источнике. В нем собрано более тридцати одной тысячи слов чувашского языка.

Процесс морфологического анализа разделен на два этапа. На первом этапе слово в исходной форме ищется в словаре основ. Грамматические характеристики в данном случае определяются по умолчанию в зависимости от части речи. На втором этапе производится непосредственный анализ слова, разбиение его на пары «корень-аффиксы» и извлечение характеристик. Оба этапа возвращают произвольное количество омонимов в зависимости от найденных совпадений. При отсутствии совпадений слово возвращается с «неопределенными» характеристиками.

Прямой поиск входного слова в словаре основ выполняет функция SearchInDictionaries.

Второй этап реализован в функции DetermineOnYourOwn. Исходное слово подвергается пошаговому разбиению на аффиксы и происходит исследование на основе выделения его компонент (основы и аффиксов) на предмет принадлежности к какой-либо части речи.

Основной задачей синтаксического блока лингвистического процессора (ЛП) является преобразование морфологической структуры (МорфС) предложения, поступающей с выхода морфологического блока, в синтаксическую структуру (СинтС). Синтаксический анализатор национального корпуса чувашского языка реализован на основе теории, изложенной в [9, с. 91–99]. Так как МорфС предложения состоит из МорфС отдельных словоформ, то переход от МорфС предложения к его СинтС осуществляется путем установления синтаксических связей между МорфС слов и между самими связями. При этом МорфС отдельных словоформ служат для установления этих связей или, как принято их называть в компьютерной лингвистике, отношений. Поэтому от того, какую модель данных для представления СинтС мы примем за основу, будет зависеть эффективность работы синтаксического блока ЛП. Структура синтаксического анализатора состоит из следующих 10 классов:

SyntParser.cs – основной класс. Содержит поля и методы для работы с библиотекой.

1. MainWork.cs – класс, который отвечает за определение характеристик предложения.

2. AttributesDeterminer.cs – класс, который предоставляет основные методы по анализу предложений. Объединяет все полученные данные под одним объектом.

3. AttrDeterHelper.cs – вспомогательный класс класса *AttributesDeterminer*.

4. Combinator.cs – класс, который на основе входного предложения формирует всевозможные вариации предложения.

5. Helper.cs – класс, в котором собраны общие вспомогательные функции.

6. BorderingWork.cs – класс, который отвечает за определение границ простых предложений в сложном.

7. Relations.cs – класс, отвечающий за определение отношений (связей) между словами.

8. GraphMaker.cs – класс, который формирует граф на основе выявленных отношений.

9. SyntConstants.cs – класс синтактических констант, используемых в проекте.

Разработанная библиотека выполняет следующие действия:

1. Определение подлежащих и сказуемых предложения.
2. Определение границ простых предложений.
3. Определение связей между словосочетаниями.
4. Визуализация выявленных связей.
5. Возможность вывода результатов в файл.
6. Режим логгирования для удобной отладки.

Определение корректности и соответствия результатов исходным данным возможно при комплексном тестировании программного обеспечения с использованием форм с характерными особенностями и исключениями в качестве входной информации [10].

Комплексное тестирование необходимо для нахождения несоответствий системы ее исходным целям. Разработанное ПО, благодаря дружественному интерфейсу, позволяет даже неопытному пользователю быстро разобраться в принципе работы системы. Главные пункты тестирования:

1. Корректность морфологического и синтаксического анализа.

2. Быстродействие программы.

Для контроля корректности анализа на вход приложения поступили слова с несколькими значениями, слова-исключения, слова с максимальным количеством букв и т.д. Далее приводятся основные направления тестирования на конкретных примерах, в статье они обозначены отдельными пунктами, результаты приведены на рисунках и в таблицах.

1. Слово «çурĕ» верно распознано. Лингвопроцессор вернул все три значения слова. Существительное «çур + ĕ» (его сруб), глагол в будущем времени «çур+ĕ» (порубит), глагол в прошлом времени «çу + р + ĕ» (помыл). Морфологические характеристики также отмечены верно (рис. 1). В программе реализовано представление нескольких значений слова, доступ к которым пользователь получает при помощи переключателя, на рис. 1 переключатель активировал позицию «1/3», данная позиция соответствует первому значению слова, последующие два значения слова определены системой корректно. Основные результаты тестирования сведены в табл. 1, в скобках указаны значения слов.

2. Слово «велосипедлисемпехчĕ» (был с велосипедными же) с шестью аффиксами протестировано корректно. Грамматические характеристики определены правильно, включая описание аффиксов (табл. 1).

Таблица 1

Результаты тестирования морфологического анализатора

Слово	Часть речи	Корень	Аффиксы	Падеж	Время	Число	Лицо
çурĕ (его сруб)	имя существительное	çур	ĕ	основной	–	единственное	3е
çурĕ (порубит)	глагол	çур	ĕ	–	будущее	единственное	3е
çурĕ (помыл)	глагол	çу	р | ĕ	–	прошлое	единственное	3е
велосипедлисемпехчĕ-и? (был с велосипедными же)	прилагательное	велосипед	ли | сем |пе |х| чĕ | -и	творительный	прошлое	множественное	–
сурăм	имя существительное	сурă	м	основной	–	единственное	1е
сурăм	глагол	су	р | ăм	–	прошлое	единственное	1е

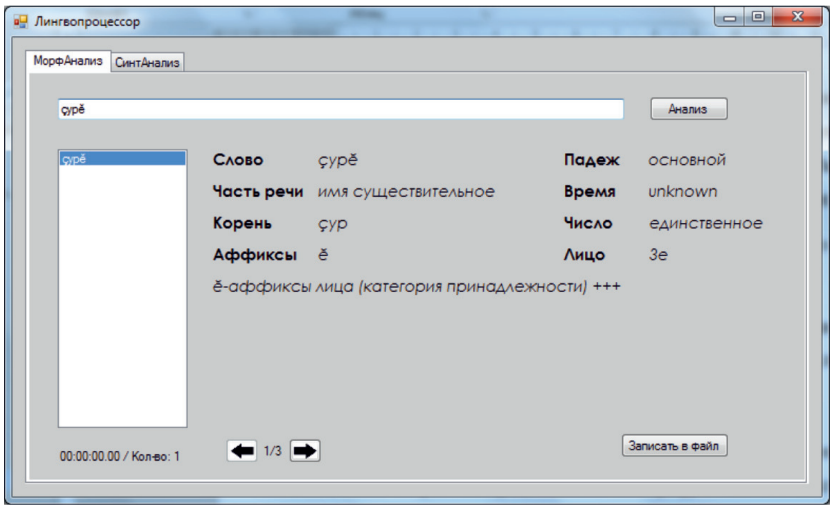

Рис. 1. Окно результатов анализа слова

3. Анализ слова «сурăм» также верно дал результаты: существительное «сурă + м» и глагол в прошлом времени «су + р + ăм».

Быстродействие оценивается средствами самого языка программирования C#. Класс Stopwatch предоставляет набор функций и свойств, которые можно использовать для точного измерения затраченного времени. Необходимо точно измерить время потраченное на анализ одного слова, нескольких слов, большого текста. В случае с синтаксическим анализатором интересует скорость анализа предложения простого и сложного [11].

В результате тестирования сделаны следующие выводы. Слова с максимальным количеством букв обрабатываются не более чем за 3 миллисекунды. 10 слов с разным количеством букв в среднем обрабатываются лингвопроцессором менее чем за 7 миллисекунд (рис. 2). Результаты анализа 100 слов с различным количеством букв следующие: система произвела анализ за 36 миллисекунд (рис. 3).

4. Анализ простого предложения «Ачасем паян шкула каяймарĕç!» завершился за 4 миллисекунды. Полученные результаты оказались корректными: типы, главные члены определены верно, граф сформировался согласно полученным связям.

5. Анализ сложного предложения. Выявлены все простые предложения, тип установлен верно, граф сформирован корректно. Время работы со сложным предложением практически не отличается от анализа простого: 3 миллисекунды (рис. 4). Результаты тестирования простых и сложных предложений сведены в табл. 2.

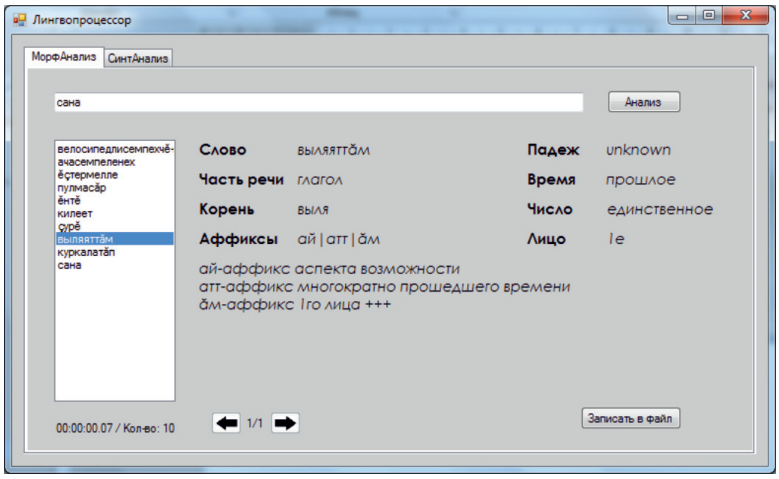

Рис. 2. Окно результатов анализа десяти слов

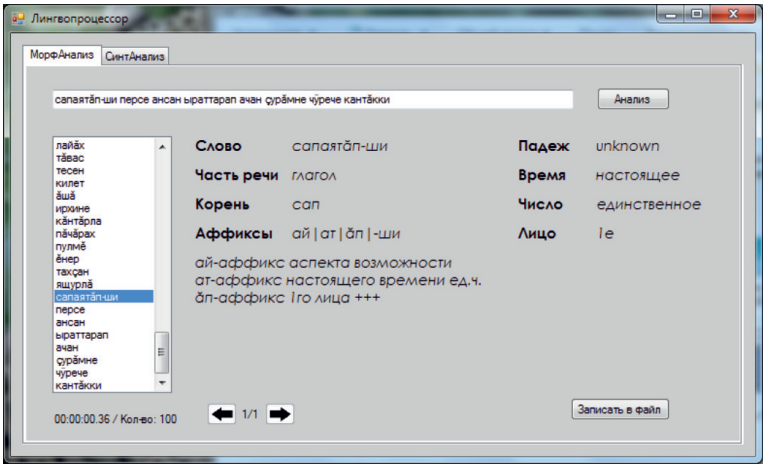

Рис. 3. Окно результатов анализа ста слов

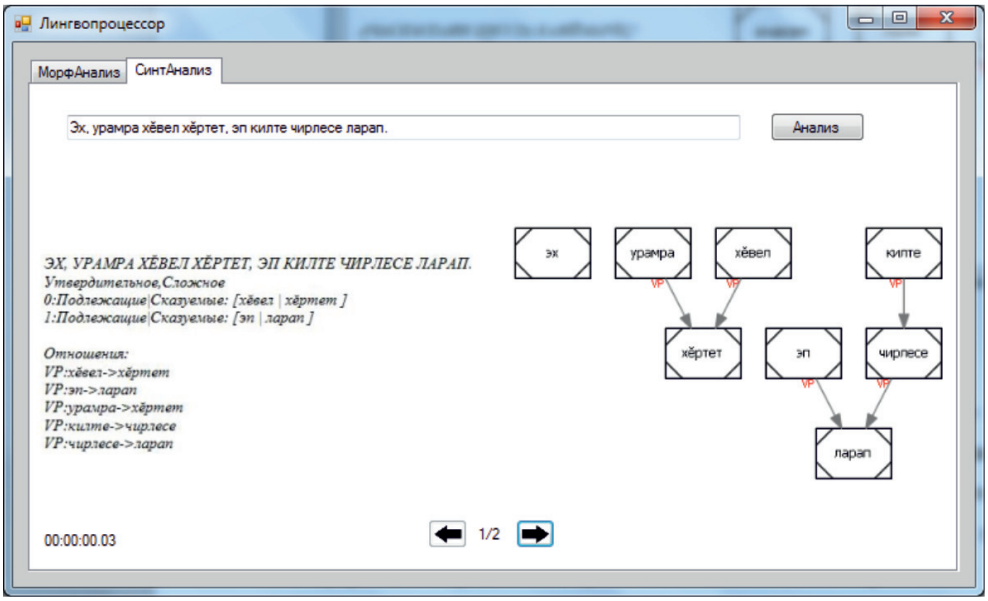

Рис. 4. Окно результатов анализа сложного предложения

Таблица 2

Результаты тестирования синтаксического анализатора

Предложение	Тип предложения, Подлежащее|Сказуемое	Отношения
Ачасем паян шкула каяймарĕç!	Восклицательное, простое 0:Подлежащее|Сказуемое: [ачасем|каяймарĕç]	VP: ачасем -> каяймарĕç VP: паян -> каяймарĕç VP: шкула -> каяймарĕç
Эх! Урамра хĕвел хĕртет, эп килте чирлесе ларап	Утвердительное, сложное 0:Подлежащее|Сказуемое: [хĕвел | хĕртет] 1:Подлежащее|Сказуемое: [эп | ларап]	VP: хĕвел -> хĕртет VP: эп -> ларап VP: урамра -> хĕртет VP: килте -> чирлесе VP: чирлесе -> ларап

Таблица 3

Результаты тестирования быстродействия системы

Входная информация	Значение	Результаты
Слово с минимальным количеством букв	ача	1 миллисекунда
Слово с максимальным количеством букв	велосипедлисемпехчĕ	3 миллисекунды
10 слов	10 слов разной длины	7 миллисекунд
100 слов	100 слов разной длины	36 миллисекунд
Простое предложение	Ачасем паян шкула каяймарĕç!	4 миллисекунды
Сложное предложение	Эх! Урамра хĕвел хĕртет, эп килте чирлесе ларап.	3 миллисекунды

Во время тестирования никаких программных ошибок не выявлено. Нажатие по кнопке «Анализ» при пустой строке входных данных не выкидывает исключений. Кнопка «Записать в файл» доступна только при наличии слов в левой колонке. Изображения всех вариаций графов и информации о предложении успешно создавались и отображались. Результаты тестирования системы на быстродействие сведены в табл. 3. Как видно из таблицы, время работы с простым или сложным предложением не превышает 3–4 миллисекунд.

В качестве эксперимента выбирались слова и предложения из разных открытых источников, размещенных на сайте национального корпуса чувашского языка. Во всех случаях при наличии морфем в словаре программа производила точный анализ с правильным определением всех атрибутов, отсутствие морфем при экспериментах не обнаружено.

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 15-04-00532.

Список литературы

1. Ганиева И.Ф. Об использовании корпусов в лингвистических исследованиях // Вестник Башкирского университета. – 2007. – Т. 12, № 4. – С. 104–106.

2. Губанов А.Р. Морфологический стандарт для систем автоматической обработки текстов на чувашском языке и архитектура грамматического словаря. В сборнике: Актуальные вопросы истории и культуры чувашского народа // Составитель и научный редактор Н.Г. Ильина; Чувашский государственный институт гуманитарных наук. – Чебоксары, 2015. – С. 146–161.

3. Губанов Д.А., Губанов А.Р. Синтаксические формальные модели русского и чувашского языков для синтаксического анализатора лингвопроцессора. В сборнике: Чтения, посвященные Дням славянской письменности и культуры // Сборник статей региональной научной конференции. – 2009. – С. 58–66.

4. Suleymanov D. Natural Cognitive Mechanisms in the Tatar language // In the Collection of the Vienna Proceedings of the Twentieth European Meeting in Cybernetics and Systems Research. Edited by Robert Trappel. Vienna, Austria, 6–9 April, 2010. – P. 210–213.

5. Ермаков А.Е. Метаязык описания грамматики в синтаксических анализаторах естественного языка на основе объектно-ориентированного языка программирования // Программная инженерия. – 2015. – № 12. – С. 12–21.

6. Захаров В.П. Корпуса русского языка. Труды института русского языка им. В.В. Виноградова. – 2015. – № 6. – С. 20–65.

7. Желтов П.В. Сравнительные исследования морфем чувашского языка. – Чебоксары: Изд-во Чуваш. ун-та, 2013. – 166 с.

8. Желтов П.В. Формальные методы и модели в сравнительно-сопоставительном языке знаний: Монография. – Чебоксары: Изд-во Чуваш. ун-та, 2006. – 252 с.

9. Желтов П.В. Лингвистические процессоры. Формальные модели и методы: теория и практика: Монография. – Чебоксары: Изд-во Чуваш. ун-та, 2006. – 208 с.

10. Майорова А.Д. Корпусная лингвистика: исторический и лингводидактический аспекты. Международный научно-исследовательский журнал. – 2017. – № 5–2 (май). – С. 42–46.

11. Шамсутдинова Э.Х., Сулейманов Р.Р. Обзор синтаксических анализаторов английского и русского языков. В сборнике: Информационные технологии в исследовательском пространстве разноструктурных языков Сборник статей I Международной интернет-конференции молодых ученых. – 2017. – С. 31–33.