элементов в коде
подогнанная к заголовку tabsize

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
Петрозаводский государственный университет (ПетрГУ)
Институт математики и информационных технологий
Кафедра прикладной математики и кибернетики

Отчет о научно-исследовательской работе

Синтаксический разбор фольклорных текстов

Выполнил:
студент 3 курса группы 22203 Е. Р. Федотова

_____
подпись

Научный руководитель:
д.т.н., доцент Л. В. Щеголева
Оценка руководителя: _____

_____
подпись

Представлен на кафедру

« ____ » _____ 2018 г.

_____
подпись принявшего работу

Петрозаводск

2018

Содержание

Введение

В наше время сфера автоматической обработки текста является актуальной. Для точной обработки текста требуется полная структура текста. Существует технология, которая помогает быстро обрабатывать информацию, а также анализировать её.

Парсинг (синтаксический анализ) в широком смысле – это автоматический анализ структуры любых текстовых данных [1]. В более понимании термин “парсинг” означает процедуру машинного анализа структуры текста на естественном языке, в том числе – структуры предложения.

Для анализа фольклорных текстов используются теоретика-графовые модели.

Теоретико-графовая модель – информационная модель определенной области, которая реализована в виде ориентированного графа, вершины которого являются объектами области, а ребра образуют связь - задают отношения между объектами области.

Для фольклорных текстов в теоретико-графовой модели в качестве вершин используются персонажи, природные явления, а также предметы быта, которым в фольклорных текстах выделена особая роль. В качестве дуг выступают действия, которые происходят между объектами.

В наши дни процесс построения теоретико-графовых моделей происходит человеком вручную. Задача научно-исследовательской работы состояла в том, чтобы автоматизировать процесс построения теоретико-графовых моделей. Эта задача связана с анализом текста с точки зрения синтаксиса.

В прошлом учебном году с данной задачей помогали справиться две технологии компании “Яндекс” - “Tomita parser” и “MyStem ”.

Однако, работа с этими технологиями не идеальна. Поэтому было принято решение, выполнить поставленую задачу с помощью другой технологии.

А именно с помощью языка программировнаия python и его библиотеки для обработки тескт nltk.

1 Теоретико-графовая модель текста

Теоретико-графовые модели текстов отображают совокупность объектов из текста в виде графа связей, также можно сказать в виде дерева зависимостей [3].
В фольклорных текстах в качестве объектов чаще всего выступают персонажи, природные явления, предметы быта и так далее. Действия в фольклорных песнях выражаются глаголами или отглагольными формами.
Зависимость между объектами можно разделить на два типа:

1. Локальный
2. Глобальный

В локальном типе связи действия выражаются глаголами и отглагольными формами, как сказано ранее, при этом они связывают объекты одного уровня, образуя синтагматические отношения в тексте.
В глобальном типе связи действия распространяются на весь текст и, как правило, никак не выражены в тексте. Такой вид связей образует парадигматические отношения между объектами.
На графе связей глобальная связь обозначается пунктирной линией, а локальная связь обозначается сплошной линией.
В качестве примера теоретико-графовые модели был взят текст “Ах, подруженьки-голубушки” Н. А. Клюева (Рис.1).

Ах, подруженьки-голубушки,
Луговые серы утушки,
Вы берите-ка скорёшенько
Пялы новые, точеные,
Еще иглы золоченые,
Шелк бурмитчатый, наводчатый,
Мелкий бисер с ясным золотом,
Расшивайте к сроку-времени
Разузорчатую завесу!
На одном углу — скради глаза,
Наведите солнце с месяцем,

На другом углу — рехнись ума,

Нижьте девушку с прилукою!

Как наедут сват со свахою,

Поезжане с девьим выкупом,

Разглядятся и раззарятся

На мудрены красны шитицы,

А раззарясь, с думы выкинут

Сватать павушку за ворона,

Ощипать перо лазорево,

Довести красу до омута!

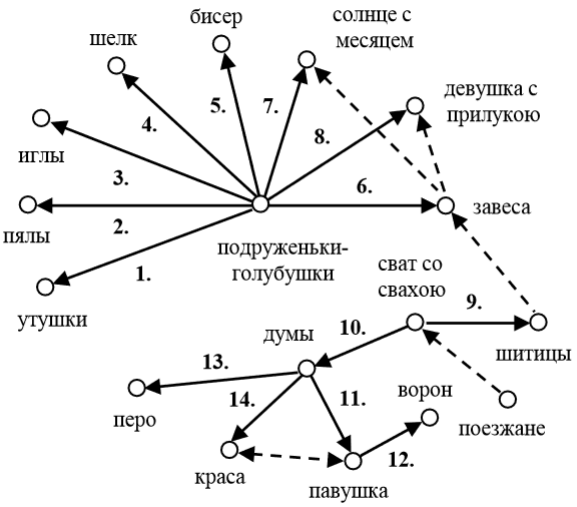

Рис. 1: Теоретико-графовая модель стихотворения “Ах, подруженьки-голубушки”

В настоящее время процесс построения графов связей происходит вручную.

Цель курсовой работы автоматизировать процесс построения теоретико-графовых моделей.

На данном этапе была поставлена задача извлечения основы предложения из фольклорных текстов, для дальнейшего построения зависимостей.

2 Обзор технологий

Для достижения цели были изучены две технологии компании “Яндекс” - “Tomita parser” и “MyStem ”. А также библиотека языка python - nltk.

2.1 Обзор Tomita parser

Томита-парсер — это инструмент для извлечения структурированных данных (фактов) из текста на естественном языке [4]. Извлечение фактов происходит при помощи контекстно-свободных грамматик и словарей ключевых слов.

Минимальный набор файлов для запуска парсера это два файла - конфигурационный файл и корневой словарь[4].

Конфигурационный файл сообщает парсеру, где искать все остальные нужные файлы, как их интерпретировать и что делать с ними. В словарях Томита-парсера содержатся ключевые слова, которые используются в процессе анализа грамматиками. Каждая статья этого словаря задает множество слов и словосочетаний, объединенных общим свойством или правилом.

Помимо двух основных файлов есть файлы для описания типов фактов, описания типов ключевых слов и грамматика. Грамматика является одним из главных файлов при извлечении фактов.

Грамматики для Томита-парсера состоят из правил и шаблонов, написанных на внутреннем языке Томита-парсера . Эти шаблоны описывают в обобщенном виде цепочки слов, которые могут встретиться в тексте. Грамматики определяют, как именно нужно представлять извлеченные факты в итоговом файле. Также Томита состоит из правил. У каждого правила есть левая и правая части, разделенных символом —>. В левой части стоит один нетерминал. В правой части стоит список терминалов или нетерминалов, после которого указываются условия, применяемые ко всему правилу в целом. Присутствие условия необязательно. Правило заканчивается точкой с запятой (;).

2.2 Обзор MyStem

Для данной работы приложение MyStem может помочь понять, как парсер распознает слово с точки зрения морфологии.

Это консольное приложение, которое производит морфологический анализ текста на русском языке. Программа работает на основе словаря и способна формировать морфологические гипотезы о незнакомых словах [5].

Запуск программы происходит через терминал, таким образом:

$ mystem {опции} {входной файл} {выходной файл}

Опции следует указывать по правилам UNIX - до имен файлов. Опции – ключи, можно склеивать и комбинировать, как и для любых программ. Стиль вывода информации на выходе зависит от ключей, которые укажет пользователь.

Но общая форма вывода такая:

Word{граммема1, граммема2, . . . }

Описание или анализ слова производится с помощью граммем.

Граммема – это грамматическое значение какой-либо характеристики слова, например: часть речи, род, число, падеж и так далее.

2.3 Обзор NLTK

Библиотека NLTK, или NLTK, — пакет библиотек и программ для символьной и статистической обработки естественного языка, написанных на языке программирования Python. Содержит графические представления и примеры данных. Сопровождается обширной документацией, включая книгу с объяснением основных концепций, стоящих за теми задачами обработки естественного языка, которые можно выполнять с помощью данного пакета [6].

В работе с nltk используется такое понятие как токенизация. Токенизация — это разбиение текста на мелкие части, токены. К токенам относятся как слова, так и знаки пунктуации. Эта полезная функция помогает разбить текст на массив важных слов. То есть после разбиения всего текста на токены, произвести чистку, например, знаков пунктуации.

С помощью nltk можно создавать грамматики, как и с помощью Tomita parser. Правила постоения грамматик и их синтаксис в nltk схож с Tomita parser. Грамматика для данной работы будет основана на категориях слов. Говоря о категориях слов, подразумевается такие категории как, именная группа (NP) или глагол (VERB) и так далее.

Благодаря такому подходу можно работать с отдельными параметрами групп, например[7]:

S->NP[CASE=nomn,NUMBER=?n,GENDER=?g] VP[NUMBER=?n,GENDER=?g]

Переменная ?g, задаёт признак GENDER(род) категории VP(глагольная группа), может обозначать как мужской, так и женский род. Однако, используя эту переменную в нескольких частях одного правила, а именно в именной группе и глагольной группе, это указывает на то что их значение должно совпадать. То есть создаем параметр согласования в роде и числе между подлежащим и сказуемым.

3 Грамматика

Была разработана грамматика для нахождения основы предложения в фольклорных текстах.

Листинг 1: Исходный файл first.cxx

```
/* Подключение кодировки для обработки естественного языка */
#encoding "utf8"
/* Описание шаблонов подлежащего: существительное в именительном падеже,
   местоимение в именительном падеже, имена, два существительных через
тире */
Pod -> Noun<GU=[nom]> | Word<GU=[SPRO,nom]> | Word<GU=[persn]> |
Noun<GU=[nom]> Hyphen Noun<GU=[nom]>;
/* Описание шаблонов сказуемого: глагол, два согласованных между собой
глагола, глагол в повелительном наклонении */
Ska -> Verb | Verb<gn-agr[1]> Verb<gn-agr[1]> | Verb<GU=[imper]>;
/* Правило отображение синтаксической структуры во множество линейно
   организованных цепочек "подлежащие...сказуемое" или
"сказуемое...подлежащие",
   интерпретация проходит без нормализации */
S -> Pod<sp-agr[1]> interp(ResultFact.Osn::not_norm) Ska<sp-agr[1]>
interp(+ResultFact.Osn::not_norm) | Ska<sp-agr[1]>
interp(ResultFact.Osn::not_norm) Pod<sp-agr[1]>
interp(+ResultFact.Osn::not_norm);
```

3.1 Описаниие грамматики

Для начала следует указать кодировку. Так как мы работаем с естественным языком, то указываем кодировку "utf8".

#encoding - указывает кодировку данного файла с грамматикой.

В общем случае основа предложения состоит из двух главных членов предложения. Этими членами предложения являются подлежащие и сказуемое.

В грамматике нетерменал “Pod” является подлежащим.

Описание шаблонов подлежащего:

1. Noun<GU=[nom]> - существительное в именительном падеже

   Терминал Noun - существительное (слово с граммемой «S»). Сюда не входят имена,

фамилии и отчества.

Помета GU (grammar union) предоставляет более широкие возможности использования граммем в грамматиках. Эта помета проверяет грамматические характеристики.

Граммема nom - номинатив, именительный падеж.

2. Word<GU=[SPRO,nom]> - местоимение в именительном падеже
   Терминал Word - любое слово, состоящее из букв русского или латинского алфавита. Также разрешаются слова записанные через дефис.
   Граммема SPRO - местоимение.

3. Word<GU=[persn]> - имена
   Граммема persn - имя.

4. Noun<GU=[nom]> Hyphen Noun<GU=[nom]> - два существительных через тире
   Терминал Hyphen - тире.

В грамматке нетерменал “Ska” является сказуемым.
Описание шаблонов сказуемого:

1. Verb - глагол
   Терминал Verb - глагол, слово с граммемой «V».

2. Verb<gn-agr[1]> Verb<gn-agr[1]> - два согласованных между собой глагола
   Помета gn-agr - согласование по роду и числу.
   “[1]” - идентификатор согласования, который указывает, какой символ с каким согласуется.

3. Verb<GU=[imper]> - глагол в повелительном наклонении
   Граммема imper - императив, повелительное наклонение.

В грамматике нетерминал “S” является правилом отображение синтаксической структуры во множество линейно организованных цепочек.
Описание шаблонов правила “S”:

1. Pod<sp-agr[1]> interp(ResultFact.Osn::not_norm)
   Ska<sp-agr[1]> interp(+ResultFact.Osn::not_norm) - цепочка “подлежащие . . . сказуемое”
   Помета sp-agr - согласование между субъектом и предикатом.
   interp - оператор для записи цепочки в поле факта интерпретации.

not_norm - интерпретация проходит без нормализации

В этом шаблоне цепочка собранная нетерминалом “S” записывается в поле Osn факта ResultFact.

“+” - оператор конкатенации, он используется в тех случаях, когда необходимо собрать в поле факта объекты из цепочек, между которыми стоят другие слова.

2. Ska<sp-agr[1]> interp(ResultFact.Osn::not_norm)

   Pod<sp-agr[1]> interp(+ResultFact.Osn::not_norm) - цепочка “сказуемое . . . подлежащие”

4 Апробация грамматики

Апробация программной системы была проведена на основе фольклорных текстов Н. А. Клюева.

В следующих разделах представлены результаты работы грамматики на трёх фольклорных текстах.

4.1 Разбор текста “Из-за леса лесу темного”

Фольклорный текст “Из-за леса лесу темного” Н. А. Клюева.

Из-за леса лесу темного,
Из-за садика зеленого
Не ясен сокол вылётывал, —
Добрый молодец выезживал.
По одёже — он купецкий сын,
По обличью — парень-пахотник.
Он подъехал во чистом поле
Ко ракитовому кустику,
С корня сламывал три прутика,
Повыстругивал три жеребья.
Он слезал с коня пеганого,
Становился на прогалине,
Черной земи низко кланяясь:
Ты ответствуй, мать-сыра земля,
С волчняком-травой, с дубровою,
Мне какой, заочно суженый,
Изо трех повыбрать жеребий?
Первый жеребий — быть лапотником,
Тихомудрым черным пахарем,
Средний — духом ожелезиться,
Стать фабричным горемыкою,
Третий — рай высокий, мысленный
Добру молодцу дарующий,

Там река течет животная,

Веют воздухи безбольные,

Младость резвая не старится,

Не седеют кудри-вихори.

В результате грамматика выделяет основы предложений и выводит их в виде таблицы (Таблица 1).

Таблица 1: Результаты разбора текста “Из-за леса лесу темного”

Результаты
Молодец выезживал
Он подъехал
Он слезал
Река течет
Веют воздухи
Седеют кудри

Сравнительный анализ ожидаемых и полученных результатов (Таблица 2).

Таблица 2: Анализ результатов разбора текста “Из-за леса лесу темного”

Основы предложений	Результаты поиска	Причина отсутствия
Сокол вылётывал	Не найдена	Парсер не может распознать слово “вылётывал”, так как это устаревшая форма слова.
Молодец выезживал	Найдена	
Он подъехал	Найдена	
Он слезал	Найдена	
Ты ответствуй	Не найдена	Местоимение “Ты” не согласуется с глаголом в повелительном наклонении
Жеребий — быть лапотником	Не найдена	Неполная грамматика
Река течет	Найдена	
Веют воздухи	Найдена	
Младость не старится	Не найдена	“Младость” не согласуется с глаголом “старится”
Не седеют кудри-вихори	Найдена, без частицы “не”, а также без второй части подлежащего	Неполная грамматика

По результатам Таблицы 2 можно сделать вывод, что грамматика работает корректно, но не точно. Грамматика охватывает 55 процентов из общего количества возможных основ предложений.

Есть три основные проблемы работы грамматики.

Первая причина это устаревшие формы слов, которые парсер не может распознать однозначно. Эту проблему удалось выявить с помощью “MyStem”, путем запуска приложения для всех файлов. И результаты таких тестов показали, что парсер может распознать не каждое слово. Но “MyStem” умеет строить гипотетические разборы для слов, не входящих в словарь[2].

Тогда вывод анализа слов происходит таким образом:

Word{Word ?= граммема1, граммема2, . . . }

Эту проблему в будущем можно устранить путем создания словаря с ключевыми словами.

Вторая проблема это несогласованность главных членов предложения.

Эту проблему в будущем можно устранить путем написания всех возможных вариантов конструкций основ предложений для фольклорных текстов.

Третья проблема “Неполная грамматика” – некоторые нестандартные варианты основ, которые на данном этапе грамматика не может обработать точно.

На основе таких проблем грамматика будет усовершенствоваться.

4.2 Разбор текста “Не огонь горит, не смола кипит”

Фольклорный текст “Не огонь горит, не смола кипит” Н. А. Клюева.

Не огонь горит, не смола кипит;
А кипит, горит ретиво сердце,
Ретиво сердце молодецкое
Ни по батюшке, ни по матушке.
А кипит, горит по красной девушке,
Что от девушки пришла весточка,
Пришла весточка, скора грамотка:
Красна девица есть трудна, больна,
Во постелюшке, во могилушке.
Я пойду с горя на почтовый двор,
Я найму пару вороных коней;
Я пойду ли ко могилушке,
Ко могилушке, ко красной девушке.

В результате грамматика выделяет основы предложений и выводит их в виде таблицы (Таблица 3).

Таблица 3: Анализ результатов разбора текста “Не огонь горит, не смола кипит”

Результаты
Огонь горит
Смола кипит
А кипит
А кипит
Пришла весточка
Пришла весточка
Девица есть
Я пойду
Я пойду

Сравнительный анализ ожидаемых и полученных результатов (Таблица 4).

Таблица 4: Результаты разбора текста “Не огонь горит, не смола кипит”

Основы предложений	Результаты поиска	Причина отсутствия
Огонь не горит	Найдена, без частицы “не”	Неполная грамматика
Смола не кипит	Найдена, без частицы “не”	Неполная грамматика
Кипит сердце	Найдена, но не точно	Несогласование главных членов предложения по роду
Горит сердце	Не найдена	Несогласование главных членов предложения по роду
Пришла весточка	Найдена	
Пришла весточка	Найдена	
Девица есть трудна	Найдена, без “трудна”	Неполная грамматика
Девица есть больна	Найдена, без “больна”	Неполная грамматика
Я пойду	Найдена	
Я найму	Не найдена	Несогласование главных членов предложения
Я пойду	Найдена	

По результатам Таблицы 4 можно сделать вывод, что грамматика работает корректно, но не точно. Грамматика охватывает 60 процентов из общего количества возможных основ предложений.

Проблемы работы грамматики такие же как и приработе с тесктом “Из-за леса лесу темного”.

4.3 Разбор текста “Все мужья до жен добры”

Фольклорный текст “Все мужья до жен добры” Н. А. Клюева.

Все мужья до жен добры,
Накупили женам тафты;
Мой муж не ласков до меня,
Не купил мне шелкова платка.
Он коровку купил,
Мне заботу снарядил.
Лучше б масла и муки купил!
Я б стряпейку наняла.
Стряпеюшка постряпывала;
А я млада похаживала,
Каблуками приколачивала.

Таблица 5: Результаты разбора текста “Все мужья до жен добры”

Основы предложений	Результаты поиска	Причина отсутствия
Муж не ласков	Не найдена	Неполная грамматика
Он купил	Не найдена	Возможная причина - наличие слова между членами предложения
Я наняла	Не найдена	Возможная причина - наличие слова между членами предложения
Стряпеюшка постряпывала	Не найдена	Парсер не может распознать слово “Стряпеюшка”, так как это устаревшая форма слова.
Я похаживала	Не найдена	Возможная причина - наличие слова между членами предложения

По результатам Таблицы 5 можно сделать вывод, что структура текста “Все мужья до жен добры” не соответствует разработанной грамматике. Грамматика охватывает 0 процентов из общего количества возможных основ предложений.

Этот показатель можно объяснить тем, что работа парсера для пользователя не видна. Поэтому нельзя точно предположить, как парсер будет работать с подлежащим, выраженным местоимением. Для устранения этой проблемы требуется более широкое изучение работы парсера.

Также помимо уже известных проблем была выявлена такая проблема как омонимия слов или лексическая неоднозначность.

Частичная лексическая омонимия – это совпадение отдельных форм слов, они называются омоформами[1]. Слово “физики” может обозначать группу ученных-физиков или быть родительным падежом слова физика.

Грамматическая омонимия – совпадение форм одного слова. Например, у слова дочь совпадают формы именительного и родительного падежей.

5 Заключение

В ходе работы были изучены две технологии компании “Яндекс”, которые были описаны во второй главе - “Tomita parser” и “MyStem”.

На основе полученных знаний и опыта при работе была построена грамматика для “Tomita parser”, которая выделяет основу предложений.

Впоследствии эта грамматика поможет при построении теоретико-графовых моделей фольклорных текстов.

В будущем планируется усовершенствовать грамматику для получения более полных результатов.

Список литературы

1. Николаев, И.С. Прикладная и компьютерная лингвистика [Текст]/ И.С. Николаев, О.В. Митренина, Т.М. Ландо.-Москва: Ленанд, 2016. - 320 с.

2. Москин, Н. Д. Сравнительный анализ структуры текстов авторских стихотворений и фольклорных песен с помощью теоретико-графовых моделей / Н. Д. Москин, А. Г. Варфоломеев, А. А. Лебедев // Компьютерная лингвистика и интеллектуальные технологии: по материалам международной конференции «Диалог 2017»

3. Москин Н. Д. Теоретико-графовые модели фольклорных текстов и методы их анализа / Н. Д. Москин; М-во образования и науки Рос. Федерации, Федер. гос. бюджет. образоват. учреждение высш. проф. образования Петрозав. гос. ун-т. - Петрозаводск : Изд-во ПетрГУ, 2013. - 147 с. : ил.

4. tech.yandex.ru[Электронный ресурс] : Электрон. дан. - [Россия] , cop. 2014-2017 . -URL : https://tech.yandex.ru/tomita/- (12.12.2017).

5. tech.yandex.ru[Электронный ресурс] : Электрон. дан. - [Россия] , cop. 2014-2018 . -URL : https://tech.yandex.ru/mystem/- (20.04.2018).

6. Natural$_L anguage_T oolkit$[] : .. – [], cop.2014 – 2018. – URL : $https$: $//ru.wikipedia.org/wiki/Natural_L anguage_T oolkit$ – (12.12.2018).$mathling.phil.spbu.ru$[] : .. – [], cop.2014 – 2018. – URL : $http$: $//mathling.phil.spbu.ru/sites/default/files/IMS2016_M OPM.pdf$ – (12.12.2018)