Прикладная лингвистика

УДК 81’322.+81’33+ 811.58
ББК Ш.11+Ш171

ИСПОЛЬЗОВАНИЕ ЛИНГВИСТИЧЕСКИ ОРИЕНТИРОВАННЫХ МОДУЛЕЙ НА ЯЗЫКЕ PYTHON ДЛЯ ОБРАБОТКИ БОЛЬШИХ ТЕКСТОВЫХ МАССИВОВ НА ВОСТОЧНЫХ ЯЗЫКАХ В ЦЕЛЯХ ЭФФЕКТИВНОГО СБОРА И ОБРАБОТКИ ДАННЫХ ПО ОТРАСЛЯМ ВОСТОКОВЕДЧЕСКОЙ ТЕМАТИКИ (НА ПРИМЕРЕ NLTK)

Б.Г. Фаткулин
Южно-Уральский государственный университет, г. Челябинск

Проведен анализ современного лингвистически ориентированного программного обеспечения, созданного в рамках языка программирования Python. В качестве примера выбран комплекс программных модулей Natural Language Toolkit (NLTK). В статье также рассматриваются не только общие принципы работы NLTK, но и их особенности в применении к восточным языкам: фарси, арабскому и китайскому. Показано решение для работы с текстами на восточных языках в кодировке utf-8.

Ключевые слова на русском языке: NLTK, восточные языки, модули Python, обработка естественных языков, код, кодировка utf-8, большие данные, UNIX.

Согласно федеральным государственным образовательным стандартам по направлению 035800.62 «Фундаментальная и прикладная лингвистика» (утверждены приказом Минобрнауки от 15.12.2010 № 1869) специалисты в рамках данной специальности должны освоить набор следующих компетенций: владеть основными методами, способами и средствами получения, хранения, переработки информации (0К-11), уметь пользоваться лингвистически ориентированными программными продуктами (ПК-16), уметь провести квалифицированное тестирование эффективности лингвистически ориентированного программного продукта (ПК-18). ФГОС также перечисляет следующие навыки: владеть методами сбора и документации лингвистических данных (ПК-9), владеть принципами создания представительных текстовых массивов, корпусов текстов, корпусов звучащей речи, мультимодальных корпусов, электронных словарей разных типов, лингвистических баз данных и уметь пользоваться этими ресурсами (ПК-15). Таким образом, вышеприведенные стандарты ориентируют как преподавательский состав, так и все категории обучаемых (специалистов, бакалавров и магистрантов) на использование в учебном процессе и в сопутствующей практической деятельности разнообразных прикладных программ, отражающих последние достижения современной информатики.

Связывая идею написания настоящей статьи в том числе и с необходимостью наполнения стандартов ФГОС практическим содержанием, мы обратили свое внимание на формулировку ПК-16 (уметь пользоваться лингвистически ориентированными программными продуктами).

В статье [3] мы обосновали необходимость использования достижений прикладной лингвистики для эффективного сбора информации на восточных языках. В глобальном информационном пространстве объемы текстов на восточных языках растут в геометрической прогрессии. Можно с уверенностью сказать, что совокупность оцифрованных текстов на восточных языках подпадает под категорию Big Data («большие данные»). В информационных технологиях понятие «большие данные» определяется как серия подходов, инструментов и методов обработки структурированных и неструктурированных данных значительных объёмов и сложного многообразия для получения воспринимаемых человеком результатов. Обработка больших массивов текстов возможностями компьютерных алгоритмов, объединенных в специальные комплексы является составной частью повышения эффективности подходов к «большим данным». Поэтому лингвистически ориентированное программное обеспечение в рамках языков программирования Perl, Java, Ruby, Haskell, Python и т. д. приобретает в последнее время все большую актуальность.

Вопросы апробации лингвистически ориентированного программного обеспечения поднимаются многими российскими лингвистами. Так, А.В. Маслов в своей статье[2], проводя обзор программного обеспечения для обработки естествен-

ных языков, упоминает и лингвистически ориентированное программное обеспечение на языке Python. Курбатов С.С., Лобзин А.П., Хахалин Г.К. в своей статье [1] проводят сравнительный анализ лингвистически ориентированного программного обеспечения по параметрам эффективности, простоты освоения, сферам применения и т. д.

Исследуя большие текстовые массивы на восточных языках, мы обратили внимание на NLTK (Natural Language Toolkit) – комплекс лингвистически ориентированных модулей на языке Python . NLTK представляет собой набор исполняемых скриптов, направленных на обработку текстовых файлов. Скрипт – это небольшая программа, как правило выполняющая одну функцию, которая как правило выполняется из командной строки. Объектом скрипта является некий текстовый файл. В результате работы скрипта выводится другой файл, содержащий результаты обработки. Комбинируя скрипты в разной последовательности, можно добиваться на выходе различных результатов. Таким образом, NLTK можно сравнить со своего рода «кулинарной книгой» прикладной лингвистики [6], а скрипты – это набор рецептов «на все случаи жизни».

Оказавшийся в фокусе нашего внимания программный комплекс был разработан в Пенсильванском университете командой прикладных лингвистов (Steven Bird, Ewan Klein, Edward Loper, and Jason Baldridge). Он применяется в университетах по всему миру как лингвистический тренажер для обучения обработке текстов на естественных языках. Можно примерно следующим образом определить сферы применения NLTK и его особенности:

1. Применимость для тренировки специалистов соответствующего профиля. Действительно, NLTK содержит множество примеров решения задач, которые стоят перед специалистами: функции, модули и субмодули NLTK охватывают все операции, которые производит прикладной лингвист. В комплекте с набором исполняемых модулей прилагается система документации, которая раскрывает значение каждого модуля. Немаловажным является и то, что система NLTK в ходе ее установки на компьютер автоматически подгружает и корпуса текстов, готовые к использованию. Сами авторы этого комплекса написали книгу, которая предназначена для потенциальных пользователей NLTK [4]. Книга содержит демонстрационный код и примеры, раскрывающие суть выполнения задач и их результаты.

2. Сложилась устойчивая группа лингвистически ориентированных программистов, которые пишут на языке Python модули для обработки не только английского, но и других языков, в том числе и восточных: арабского, фарси, китайского и т. д. Программисты пишут исполняемые модули и размещают их в специальных онлайн репозиториях. Любой желающий может ознакомиться с кодом программы и либо просто использовать его, либо настроить его для выполнения своих специфических задач. Это означает, что достижениями nltk могут воспользоваться и востоковеды, прикладные лингвисты, работающие в области восточных языков, архивного востоковедения и востоковедческой аналитики [3].

Приведенные выше тезисы подтверждают вывод о том, что NLTK содержит инструментарий для выполнения большинства задач компьютерной лингвистики. Немаловажно, что этот комплект программ доступен пользователям на правах GNU (General Public License). Код программ является открытым, следовательно пользователи могут добавлять свой код, совершенствуя его. Документация к NLTK содержит автоматически генерируемый каталог для каждого класса, метода и функции, а также несколько отдельных руководств для выполнения специализированных задач. Хотя в сравнении с промышленными программами, распространяемыми на платной основе, комплекс NLTK может показаться достаточно медленным и сложным для освоения, однако он может стать средством для получения нетривиальных результатов.

Целью нашего исследования стала апробация NLTK в применении к языкам использующим нетрадиционные формы представления текстов (арабская вязь и китайская иероглифика). Для достижения цели мы наметили выполнить следующие задач:

1) найти и апробировать модули Python, работающие в сфере обработки текстов на восточных языках;

2) решить проблему с отображением текстов в кодировке utf-8; (Материалом исследования стали тексты на восточных языках, содержащие актуальную информацию по состоявшемуся в сентябре 2014 г. IV Саммиту лидеров прикаспийских государств).

Список модулей и субмодулей NLTK вызывается командой:

```
Python 2.7.6 (default, Mar 22 2014, 22:59:38)
[GCC 4.8.2] on linux2
>>> import nltk
>>> help(nltk)
```

На момент времени “Ср. окт. 22 10:23:48 YEKT 2014” NLTK содержал 38 модулей и 66 субмодулей, выполняющих задачи прикладной лингвистики.

Проанализировав вывод данной команды, мы пришли к выводу, что модули NLTK действительно обеспечивают выполнение стандартных задач по обработке текстовой информации на основных европейских языках. Для выполнения нестандартных функций, в частности для обработки текстов на восточных языках, необходимо искать в репозиториях специальные модули Python, загружать, устанавливать и апробировать их.

Официальная страница Python содержит ссылку на репозиторий открытого программного кода. Пользователи языка Python могут использовать этот код для решения своих задач, создавать

свой собственный код, регистрироваться как участники и размещать свой код на личной странице для всеобщего использования. Поисковые запросы к содержимому репозитория подтвердили наличие специальных ветвей «Chinese», «Persian», «Arabic». Множества пакетов разработано и для работы с русским языком (ветвь «Russian»).

В качестве примера приведем полученный нами список пакетов для работы с языком фарси:

1. persian-0.0.3 12 Persian.py: A simple Python library for Persian language localization
2. Khayyam-0.9.2 5 Khayyam (Jalali Persian Datetime) library
3. hazm-0.3 3 Python library for digesting Persian text.
4. negar-cli-1.0.1 3 Negar Command Line Interface.

Дальнейшие запросы также вывели обширный список модулей для арабского и китайского языков.

Как мы уже сказали, существенной чертой NLTK является то, что он может работать и с восточными языками, использующими нетрадиционные средства представления, в том числе арабскую вязь и иероглифы. Новый шаг в развитии обработки текстов на восточных языках был сделан после изобретения и введения в употребление кодировки utf-8, которая дала возможность выводить на экран большинство символов текстово-символьной таблицы наиболее значимых языков мира. Однако без предварительной подготовки вывод результатов команд Python не всегда отображается корректно, поскольку символы восточных языков относятся к кодировке utf-8, чего не учитывают настройки обычного пользователя, данные ему по умолчанию.

Предлагаем скрипт для решения этой проблемы в применении к тексту на фарси:

```
# -*- coding: utf-8 -*-
import codecs
file = codecs.open("/home/lisboa/text-na-farsi.txt","r","utf-8")
text = file.read()
print text
text = text[0:]
print text
words = text.split()
print words
for w in words:
   print w
```

Исполнив данный скрипт, мы в итоге получили удобочитаемый текст на одном из восточных языков по выбору: это мог быть китайский, фарси или арабский. Этот текст мы можем подвергать дальнейшей обработке средствами языка Python.

Полученные нами результаты мы интерпретируем следующим образом:

1. В ходе выполнения учебных планов по специальности «Фундаментальная и прикладная лингвистика» целесообразно использовать Python и его модули, в частности NLTK.

2. Лингвистически ориентированное программное обеспечение может использоваться не только в учебных целях, но и в работе специализированных востоковедческих структур, проводящих мониторинг информации на восточных языках.

3. Лингвистически ориентированное программное обеспечение должно войти в рабочий арсенал российских лексикографов и терминографов как инструмент для извлечения терминологии и другой значимой информации в специальных предметных областях.

Литература

1. Курбатов, С.С. Программное обеспечение для лингвистически-ориентированного пополнения онтологии: докл. / С.С. Курбатов, А.П. Лобзин, Г.К. Хахалин // Четырнадцатая конференция по искусственному интеллекту с международным участием. – Казань, 2014. – Т. 3. – С. 164–172.

2. Маслов, А.В. Системы автоматической обработки текстов на естественном языке: лингвистические аспекты и перспективы развития / А.В. Маслов // Вестник Московского государственного лингвистического университета. – 2013. – № 13 (699). – С. 167–170.

3. Фаткулин, Б.Г. Прикладная лингвистика и обработка текстов на восточных языках: современные перспективы / Б.Г. Фаткулин // Вестник ЮУрГУ. Серия «Лингвистика». – 2014. – Т. 11, № 3. – С. 15–18.

4. Bird, S. Natural language processing with Python. / S. Bird, E. Klein, and E. Loper. – Beijing; Cambridge; Mass: O'Reilly, 2009. Print.

5. Garrette, D. An extensible toolkit for computational semantics / D. Garrette, E. Klein // Proceedings of the Eighth International Conference on Computational Semantics (IWCS-8 '09) / H. Bunt, V. Petukhova, S. Wubben (Eds.). Association for Computational Linguistics, Stroudsburg, PA, USA, 2009. – P. 116–127.

6. Perkins, J. Python text processing with NTLK 1.0 Cookbook over 80 practical recipes for using Python's NLTK suite of libraries to maximize your natural language processing capabilities / J. Perkins. – Birmingham Mumbai: PACKT Publishing, 2010. Print.

Фаткулин Булат Гилимдарович, кандидат филологических наук, доцент кафедры общей лингвистики, Южно-Уральский государственный университет (Челябинск), bfatkulin@gmail.com

Поступила в редакцию 2 ноября 2014 г.

USE OF THE LINGUISTICALLY ORIENTED PYTHON LANGUAGE MODULES FOR HANDLING LARGE TEXTS IN THE EASTERN LANGUAGES IN ORDER TO MINE THE ORIENTALISTICS DATA (WITH NLTK MODULE TAKEN AS AN EXAMPLE)

***B.G. Fatkulin**, South Ural State University, Chelyabinsk, Russian Federation, bfatkulin@gmail.com*

This article analyzes the contemporary linguistically oriented software created on the basis of the programming language Python. The Natural Language Toolkit (NLTK) is selected as an example. The research considers not only the general principles of the NLTK but also the principles especially applied to the eastern languages: Farsi, Arabic and Chinese. The author shows certain solutions for work with texts in Unicode as input-output for Python text processing modules.

Keywords: NLTK, eastern languages, modules, Python, natural language processing, code, encoding utf-8, big data, UNIX.

References

1. Kurbatov S.S., Lobzin A.P., Hahalin G.K. Programmnoe obespechenie dlya lingvisticheski-orientirovannogo popolneniya ontologii. [Software for the Linguistically oriented Ontology Replenishment] *The Proceedings of 14 International Conference on the Artificial Intellect*. Kazan', 2014, iss. 3, pp. 164–172.

2. Maslov A.V. [The Systems of the Automatic Processing of Natural Language Texts, Linguistic Aspects and Prospects of Development]. *Bulletin of Moscow State Linguistic University*. 2013, no. 13 (699). pp. 167–170. (in Russ.)

3. Fatkulin B.G.. [Applied Linguistics and Text Processing Oriental Languages: Contemporary Perspectives]. *Bulletin of the South Ural State University. Ser. Linguistics*. 2014, vol. 11, no. 3. pp. 15–18. (in Russ.)

4. Bird Steven, Ewan Klein, and Edward Loper. *Natural Language Processing with Python*. Beijing; Cambridge, Mass, O'Reilly, 2009. Print.

5. Dan Garrette and Ewan Klein. 2009. An Extensible Toolkit for Computational Semantics. *In Proceedings of the Eighth International Conference on Computational Semantics* (IWCS-8 '09), Harry Bunt, Volha Petukhova, and Sander Wubben (Eds.). Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 116–127.

6. Perkins Jacob. *Python text processing with NTLK 1.0 Cookbook over 80 practical recipes for using Python's NLTK suite of libraries to maximize your natural language processing capabilities.* Birmingham; Mumbai: PACKT Publishing, 2010. Print.

Received 2 November 2014

БИБЛИОГРАФИЧЕСКОЕ ОПИСАНИЕ СТАТЬИ

Фаткулин, Б.Г. Использование лингвистически ориентированных модулей на языке Python для обработки больших текстовых массивов на восточных языках в целях эффективного сбора и обработки данных по отраслям востоковедческой тематики (на примере NLTK) / Б.Г. Фаткулин // Вестник ЮУрГУ. Серия «Лингвистика». – 2015. – Т. 12, № 1. – С. 72–75.

REFERENCE TO ARTICLE

Fatkulin B.G. Use of the Linguistically Oriented Python Language Modules for Handling Large Texts in the Eastern Languages in order to Mine the Orientalistics Data (with NLTK Module Taken as an Example). *Bulletin of the South Ural State University. Ser. Linguistics.* 2015, vol. 12, no. 1, pp. 72–75. (in Russ.)