УДК 81’33

Джунковский Андрей Владимирович
кандидат филологических наук,
заведующий кафедрой прикладной
и экспериментальной лингвистики ФАЯ,
старший научный сотрудник
экспериментально-фонетической лаборатории
криминалистики по речеведению
Московский государственный
лингвистический университет
г. Москва, Россия

Andrey Dzhunkovskiy
PhD in Philology, Head of Applied
and Experimental Linguistics Department,
Senior Researcher at Experimental
Phonetics and Forensic Linguistics
Laboratory
Moscow State Linguistic University
Moscow, Russia
Vetinari01@gmail.com

Изюмская-
Капитонова Вероника Викторовна
младший научный сотрудник лаборатории
корпусной лингвистики,
преподаватель кафедры прикладной
и экспериментальной лингвистики ФАЯ
Московский государственный
лингвистический университет
г. Москва, Россия

Veronika Izyumskaya-Kapitonova
Junior Researcher at Corpora Laboratory,
Lecturer of Applied
and Experimental Linguistics
Department
Moscow State Linguistic University
Moscow, Russia
izyumskayaw@gmail.com

Мелина Александра Юрьевна
младший научный сотрудник
экспериментально-фонетической лаборатории
криминалистики по речеведению
Московский государственный
лингвистический университет
г. Москва, Россия

Alexandra Melina
Junior Researcher at Experimental
Phonetics and Forensic
Linguistics Laboratory
Moscow State Linguistic University
Moscow, Russia
mcshane97.pm@gmail.com

ВОЗМОЖНОСТИ РЕШЕНИЯ ЗАДАЧ ЛИНГВИСТИЧЕСКОЙ ЭКСПЕРТИЗЫ ТЕКСТА КОМПЬЮТЕРНЫМИ МЕТОДАМИ

В статье представлены возможности применения методов компьютерной лингвистики для решения задач лингвистической экспертизы текста. Авторы рассматривают способы определения лексического состава текстов и автоматического формирования частотно-ранговых распределений с последующим сравнением этих данных для анализа

оригинальности/производности текстов. Предложенный алгоритм и разработанный на его основе код могут быть эффективно использованы не только в указанных случаях, но и для решения других научных и практических задач в области лингвистики. Их преимущества включают высокую точность, скорость обработки и возможность автоматизации процесса, что делает их ценным инструментом для исследователей и практиков в области языкознания. Такой подход позволяет значительно сократить время и усилия, затрачиваемые на анализ больших объемов текстовой информации, и сделать процесс лингвистической экспертизы более результативным и надежным. Авторы статьи разрешают использовать указанный код в научных, образовательных и коммерческих целях при условии ссылки на статью.

К л ю ч е в ы е с л о в а: *лингвистическая экспертиза; компьютерная лингвистика; Python; NLTK; лемматизация.*

COMPUTER TOOL UTILITY IN FORENSIC LINGUISTICS

In this article we present some possibilities of using methods of computational linguistics for the purposes of forensic linguistic text examination. Authors describe ways of determining text lexical composition and automatically creating frequency-rank distributions. This is followed by comparing the resulting data for different texts to determine their originality. We propose an algorithm and corresponding code that may be efficiently used not only in the particular case we focus on, but for a wide variety of other research and practical tasks in the field of linguistics. Their advantages include high accuracy, processing speed, and the inherent ability to automate the process, making them a valuable tool for researchers and practitioners in the field of linguistics. This approach significantly reduces the time and effort required for analyzing large volumes of textual information, making some aspects of forensic linguistic examination more reliable and effective. The authors of the article permit the use of the provided code for scientific, educational, and commercial purposes, under the condition of referencing the article.

K e y w o r d s: *forensic linguistic examination; computational linguistics; Python; NLTK; lemmatization.*

Под лингвистической экспертизой текста подразумевается зафиксированное в устном или письменном виде мнение лингвиста о каком-либо продукте речевой способности человека, используемом в коммуникации между людьми либо в общественной коммуникации между политическими субъектами. Лингвистическая экспертиза текста является направлением прикладной лингвистики и охватывает множество сфер практического применения, включая юриспруденцию, историю, литературоведение.

Принимая за критерий изучаемый объект, лингвистические экспертизы текста можно разделить на:

1) экспертизы звучащей речи, применяющие акустику, фонетику и фонологию с целью установления автора речевого высказывания и решения иных вопросов;

2) экспертизы письменного текста, использующие морфологический, синтаксический, семантический и лингвостатический анализ для достижения широкого ряда поставленных перед экспертом целей;

3) вербально-визуальные экспертизы, при проведении которых исследуется сочетание вербальной (текста) и невербальной (изображения) информации.

Методика проведения лингвистической экспертизы текста включает в себя определение следующих критериев:

1) объект анализа (установление границ лингвистического аспекта поставленной перед экспертом задачи);

2) круг задействованных источников, возможное ограничение его объемов;

3) совокупность источников лингвистической информации, теоретической и нормативной базы;

4) потенциал применения машинной обработки лингвистических данных;

5) нормативно-правовая база (законы, подзаконные акты и др.).

При проведении лингвистической экспертизы текста могут рассматриваться практически все разделы теоретического языкознания, однако наибольший интерес в этом отношении представляет лингвопрагматика. В частности, в лингвистической экспертизе текста активно применяется теория речевых актов, теория речевого воздействия, а также теория аргументаций. Таким образом, перед лингвистами находится широкий спектр задач, которые потенциально возможно решить, применяя компьютерные методы [1, с. 10–19].

Одной из задач современной лингвистической экспертизе текста является решение вопроса об оригинальности текста. Инициаторы таких лингвистических исследований часто обращаются к экспертам-лингвистам в рамках судебных разбирательств по вопросам защиты авторских прав [2, с. 61]. Типичные задачи, стоящие перед лингвистами в таких случаях, заключаются в установлении оригинальности текста, а также решении вопроса о том, является ли один текст производным произведением другого текста.

Решение этих задач в современном мире строится на методике проведения лингвистической экспертизы и может дополняться программным обеспечением, которое позволяет внести элемент автоматизации в работу эксперта-лингвиста. При этом широко распространенные программы выявления заимствований, такие как Антиплагиат, ReText.AI, Руконтекст и иные, удовлетворительно справляются с задачей выявления внешних заимствований (хотя их использование сопряжено с большим количеством споров в научной среде [3, с. 40–45]), но не способны сравнить два и более текстов друг с другом.

В этих случаях эксперт может прибегнуть к таким инструментам, как Copyleaks, Kaleidoscope или ABBYY Finereader, однако полученные результаты недостаточны для проведения глубокого анализа. Основной недостаток заключается в том, что эксперт-лингвист часто прибегает к методам квантитативной лингвистики и стремится извлечь количественные данные о текстах, что является одним из важных аспектов компаративного анализа текстов при проведении лингвистической экспертизы такого рода [4, с. 244].

В нашем опыте лингвистической экспертизы было выполнено исследование, в ходе которого нами был создан алгоритм, основанный на методах компьютерной лингвистики.

В рамках исследования лексики спорных текстов была выполнена процедура построения частотно-рангового распределения лексем анализируемого материала [5, с. 125–128].

Первый этап работы включал преобразование исходного файла формата docx в текстовый формат (plain text) с расширением .txt – текстовый файл строкового типа, лишенный большинства элементов форматирования (шрифт, кегль, таблицы и т.п.) за исключением табуляции и переноса на новую строку, которые реализуются непосредственно в кодировке файла с помощью специальных символов («\t» и «\n» соответственно). Необходимость описываемого преобразования обусловлена главным образом дальнейшим использованием языка программирования Python, в рамках которого обработка файлов формата docx требует установки дополнительных библиотек. Результат преобразования был закодирован посредством 8-битного формата преобразования Юникода (Unicode Transformation Format, 8-bit; UTF-8), что гарантирует адекватность прочтения и обработки кириллических символов.

Второй этап заключался в непосредственном написании компьютерного кода на языке программирования Python версии 3.12 в интегрированной среде разработки PyCharm Community Edition 2023.3.3. и включал в себя следующие основные шаги:

1. Импорт внешних библиотек и модулей. В рамках описываемого алгоритма использовались collections.Counter для подсчета частотности слов, для работы с регулярными выражениями, nltk (nltk.corpus.stopwords) для удаления стоп-слов, а также pymystem3 для лемматизации текста. Важно упомянуть, что библиотека nltk, изначально разработанная на материале английского языка, также часто используется для целей лемматизации, однако pymystem3, будучи созданной отечественной компанией «Яндекс», показывает большую эффективность и точность при работе с русскоязычным текстом.

2. Определение стоп-слов. Модуль nltk.corpus.stopwords содержит в себе словоформы, которые, по мнению разработчиков, являются наиболее частотными и общеупотребительными, вследствие чего они признаются наименее информативными при анализе отличительных особенностей отдельного самостоятельно текста. Так, для русского языка список стоп-слов включает служебные слова, местоимения, числительные *один*, *два*, *три*, некоторые наречия (например, *сейчас*) и т.п. Однако предложенный алгоритм позволяет также расширять список стоп-слов самостоятельно, если это представляется необходимым в результате первичного экспертного анализа текста.

3. Предобработка текста. Представляет собой блок процедур, который включает:

а) приведение всех слов текста к нижнему регистру, т.к. в рамках работы с Python строки типа *Словоформа* и *словоформа* являются самостоятельными единицами, соответственно, при дальнейшем частно-ранговом распределении их квантитативные показатели будут считаться отдельно;

б) удаление с помощью регулярных выражений всех символов, кроме буквенных, что позволяет снизить уровень шумов (в частности, цифры, отражая фактологическую информацию, не несут в себе сведений об идиостиле, который зачастую является объектом экспертного исследования);

в) лемматизацию, под которой понимается процесс автоматического приведения всех словоформ текста к их леммам (словарным формам); данная процедура позволит избежать разбиение единиц типа *словоформы* и *словоформу* на отдельные элементы в рамках частотно-рангового распределения и приведет их к общей единице *словоформа*;

г) удаление стоп-слов, которые были выделены на предыдущем этапе.

Данный список процедур предобработки не является исчерпывающим и может включать дополнительные элементы, которые обусловлены задачами экспертного исследования, например, возможно удаление единиц текста, которые не превышают заданного порогового значения по количеству символов, что также позволит сократить количество служебных слов, которые не входят в стандартный модуль nltk.corpus.stopwords.

4. Частотное распределение и ранжирование. Включает в себя использование модуля collections.Counter для подсчета частотности каждой единицы предобработанного текста, а также их сортировку согласно убыванию данного квантитативного показателя и присвоение соответствующего ранга. Результат частотно-рангового распределения может быть выведен на экран внутри среды разработки и сохранен в формате таблицы.

```
        #Обращаемся к внешним библиотекам
from collections import Counter
import re
from pymystem3 import Mystem
import nltk

from nltk.corpus import stopwords

mystem = Mystem()

file_path = "C:/Текст1.txt"
#добавляем стоп-слова
additional_stopwords = ['', '',… ]
#объединяем внешний список стоп-слов с нашим
stop_words = stopwords.words('russian') + additional_stopwords
#читаем файл
with open(file_path, 'r', encoding='utf-8') as content:
  text_to_analyze = content.read()

#чистим текст
def clean_text(text):
  # переводим в нижний регистр
  text = text.lower()
```

```
  # убираем спецсимволы
  text = re.sub('[^а-яА-Яа-zA-ZёЁ]', ' ', text)
  text = re.sub(r'\s+', ' ', text)

  # Фильтрация стоп-слов
  text = ' '.join([word for word in text.split() if word not in stop_words])

  # Лемматизация текста с помощью MyStem
  lemmas = mystem.lemmatize(text)
  text = ' '.join(lemmas)
  # убираем слова короче 3 букв
  text = ' '.join([word for word in text.split() if len(word) > 2])
  text = text.strip()
  return text

#создаем частотно-ранговое распределение
cleaned_text = clean_text(text_to_analyze)
words = cleaned_text.split()
word_frequencies = Counter(words)
ranked_words = sorted(word_frequencies.items(), key=lambda x: x[1], reverse=True)
#выводим список
for I, (word, freq) in enumerate(ranked_words, 1):
  print(f'{i}. {word} – {freq} раз.»)
```

Результатом использования этого кода стало частотно-ранговое распределение для двух текстов в формате:

Порядковый номер. Слово – кол-во употреблений.

Сравнение получившихся в результате работы кода таблиц позволило сделать выводы о лексической схожести различных текстов, что может значительно упростить работу эксперта-лингвиста при решении задачи определения оригинальности/признаков переработки текста.

ЛИТЕРАТУРА

1. Баранов А. Н. Лингвистическая экспертиза текста. Теоретические основания и практика. М. : Флинта, 2007. 594 с.

2. Как провести лингвистическую экспертизу спорного текста? Памятка для судей, юристов СМИ, адвокатов, прокуроров, следователей, дознавателей и экспертов / Под ред. проф. М. В. Горбаневского. М. : Юридический Мир, 2006. 112 с.

3. Серго А. Г. «Антиплагиат» и другие средства снижения качества научных текстов // Журнал Суда по интеллектуальным правам. 2023. № 4 (42). С. 40–45.

4. Типовые экспертные методики исследования вещественных доказательств. Ч. I / под ред. канд. техн. наук Ю. М. Дильдина. М. : ЭКЦ МВД России, 2010. 568 с.

5. Долинский В. А. Семейство ранговых распределений в квантитативной лингвистике // Вестник МГЛУ. Гуманитарные науки. 2018. № 6 (797). С. 124–155.