ПРОБЛЕМЫ ОБЩЕГО И ТИПОЛОГИЧЕСКОГО ЯЗЫКОЗНАНИЯ

УДК 81’33

Горожанов Алексей Иванович
доцент, доктор филологических наук, профессор кафедры грамматики и истории немецкого языка
Федеральное государственное бюджетное образовательное учреждение высшего образования «Московский государственный лингвистический университет»
г. Москва, Россия

Гусейнова Иннара Алиевна
доцент, доктор филологических наук, профессор кафедры лексикологии и стилистики немецкого языка
Федеральное государственное бюджетное образовательное учреждение высшего образования «Московский государственный лингвистический университет»
г. Москва, Россия

Степанова Дарья Валерьевна
кандидат филологических наук, доцент, доцент кафедры теории и практики английской речи
Минский государственный лингвистический университет
г. Минск, Беларусь

Alexey Gorozhanov
Associate Professor, Habilitated Doctor of Philology, Professor at the Department of German Grammar and History
Moscow State Linguistic University
Moscow, Russia
a_gorozhanov@mail.ru

Innara Guseinova
Associate Professor, Habilitated Doctor of Philology, Professor at the Department of German Lexicology and Stylistics
Moscow State Linguistic University
Moscow, Russia
ginnap@mail.ru

Darya Stepanova
PhD, Associate Professor, Associate Professor of the Department of English Theory and Speech Practice
Minsk State Linguistic University
Minsk, Belarus
daryastepanova79@gmail.com

СТАНДАРТИЗИРОВАННАЯ ПРОЦЕДУРА ПОЛУЧЕНИЯ СТАТИСТИЧЕСКИХ ПАРАМЕТРОВ ТЕКСТА (на материале цикла рассказов Дж. Лондона «Смок белью. Смок и малыш»)

STANDARDIZED PROCEDURE FOR OBTAINING STATISTICAL PARAMETERS OF A TEXT (on the material of the stories by J. London “Smoke Bellew. Smoke and Shorty”)

Статья посвящена проблеме интерпретации художественного произведения точными методами. Рассматривается возможность и эффективность использования современных программных инструментов для определения статистических параметров аутентичных текстов. Написанные на языке программирования Python коды с применением библиотеки обработки естественного языка spaCy позволили разработать процедуру нормализации текста и получить количественные данные о заданных параметрах текста для анализа его содержания.

К л ю ч е в ы е с л о в а: *нормализация текста*; *токен*; *библиотека spaCy*; *идиостиль*; *статистические параметры текста*; *программный код*.

The article deals with the problem of application of precise methods of linguistic research to fiction interpretation. The article examines the possibility and efficiency of using modern software tools to determine the statistical parameters of the unmarked original fictional texts. The developed Python library programs based on the spaCy natural language processing allow the authors of the article to develop the procedure of text normalization and to obtain the frequencies of the given text parameters to analyze its content.

K e y w o r d s: *text normalization*; *token*; *spaCy library*; *ideostyle*; *statistical parameters of the text*; *programming code*.

Творчество Дж. Лондона до сих пор интересует многих лингвистов и литературоведов, а также специалистов в области лингводидактики, которые видят в текстах писателя большой потенциал для анализа и практического использования в своих профессиональных сферах [1; 3; 4].

В представленной работе мы преследуем цель с помощью современных программных инструментов проанализировать цикл рассказов Дж. Лондона «Смок Белью. Смок и Малыш» для фиксации статистических параметров аутентичных текстов, что впоследствии поможет провести исследование, направленное на получение параметров идиостиля автора на синтаксическом и семантическом уровнях.

Для достижения поставленной цели необходимо решить следующие задачи:

1) стандартизировать процедуру нормализации текста для эффективной работы с помощью библиотеки обработки естественного языка spaCy;

2) получить программным способом общие статистические параметры исследуемого массива текстов;

3) получить программным способом данные о синтаксической и частеречной характеристике текста;

4) выявить некоторые идиостилистические языковые единицы в исследуемом тестовом массиве.

Методами исследования являются автоматический и автоматизированный программный анализ, а также синтез полученных данных.

Для решения первой задачи был выбран язык программирования высокого уровня Python, который на текущий момент является мировым лидером по популярности благодаря своей универсальности и наличию большого набора библиотек для решения различных прикладных задач, например, библиотека spaCy, представляющая собой набор разнообразных инструментов для расширенной обработки естественного языка. На языке Python был разработан программный код, который можно применять для работы практически с любым электронным текстом формата TXT, в том числе полученным из файла PDF. Под нормализацией здесь мы понимаем трансформацию текста в такой вид, который был бы удобен для автоматической обработки с помощью специализированного программного обеспечения. Этот процесс проводился в несколько этапов:

1) замена двойных символов переноса строки на одинарный символ переноса строки;

2) замена символов переноса строки на единичные пробелы;

3) удаление серий из двух и более пробелов с заменой их на одинарный пробел (листинг 1):

```
# Нормализация
f = open("London/LondonAll.txt", "r", encoding="utf-8")
text = f.read()
f.close()
text = text.replace("\n\n", "\n")
text = text.replace("\n", " ")
while "  " in text:
    text = text.replace("  ", " ")
f = open("London/LondonAll01.txt", "w", encoding="utf-8")
f.write(text)
f.close()
```

Листинг 1. Код универсальной программы нормализации текста

В результате работы приведенной программы осуществляется трансформация исходного текста фактически в один абзац, что представляется целесообразным в свете решения последующих задач исследования.

В рамках решения второй задачи исследования с помощью библиотеки обработки естественного языка spaCy были получены общие статистические параметры текста, под которыми понимаются данные о количестве токенов и предложений в тексте (листинг 2):

```
# Определение количества токенов и предложений
f = open("London/LondonAll01.txt", "r", encoding="utf-8")
text = f.read()
f.close()
doc = nlp(text)
print(len(doc))
for index, sent in enumerate(doc.sents):
   pass
print(index+1)
```

Листинг 2. Фрагмент кода программы для получения общих статистических данных

В результате в тексте было выявлено 119 962 токена и 8 012 предложений. Здесь под *токеном* понимаются единицы, полученные в ходе работы токенизатора spaCy, которые преимущественно представлены словами и знаками пунктуации. На предложения текст также разделяется по внутренним правилам spaCy. Как правило, в качестве формальных разделителей здесь выступают точки, а также вопросительные и восклицательные знаки.

В результате решения третьей задачи исследования с помощью библиотеки spaCy были получены данные о синтаксической и частеречной характеристике текста. Прежде всего нами был выявлен частеречный состав токенов (листинг 3):

```
# Частиречи
f = open("London/LondonAll01.txt", "r", encoding="utf-8")
text = f.read()
f.close()
doc = nlp(text)
partsList = []
for token in doc:
   if token.pos_ not in partsList:
      partsList.append(token.pos_)
print(partsList)
```

Листинг 3. Фрагмент кода программы для получения списка частей речи

Заметим, что под *частями речи* здесь понимаются особые категории spaCy, которые только частично совпадают с привычным для лингвистов списком частей речи (приводятся принятые spaCy сокращения и их расшифровка):

PUNCT (punctuation); NOUN (noun); VERB (verb); PRON (pronoun); ADP (adposition); DET (determiner); AUX (auxiliary); ADJ (adjective); ADV (adverb); PROPN (proper noun); CCONJ (coordinating conjunction); PART (particle); SCONJ (subordinating conjunction); NUM (numeral); INTJ (interjection); X (other; т.е. spaCy не смог классифицировать токен); SYM (symbol).

Далее были уточнены количественные показатели:

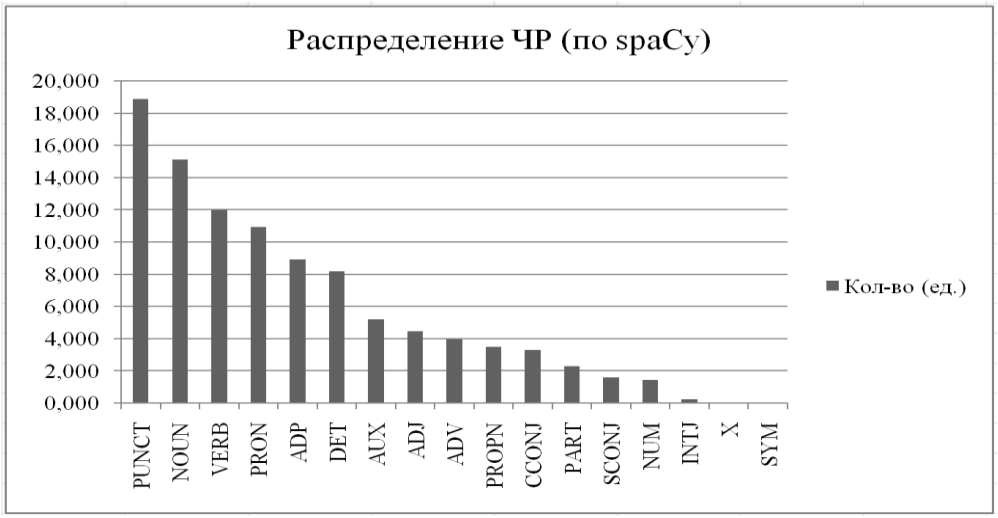
PUNCT : 22665; NOUN : 18119; VERB : 14391; PRON : 13104; ADP : 10683; DET : 9831; AUX : 6246; ADJ : 5366; ADV : 4744; PROPN : 4205; CCONJ : 3933; PART : 2748; SCONJ : 1927; NUM : 1701; INTJ : 282; X : 13; SYM : 4.

Указанный результат был получен в ходе исполнения следующей программы (листинг 4):

```
# Частиречи,количество
f = open("London/LondonAll01.txt", "r", encoding="utf-8")
text = f.read()
f.close()
doc = nlp(text)
partsList = {}
for token in doc:
   if token.pos_ not in partsList.keys():
      partsList[token.pos_] = 1
   else:
      num = partsList[token.pos_]
      partsList[token.pos_] = num+1
uniqueAll = []
for i in sorted(partsList.items(), key=lambda kv: kv[1], reverse = True):
   uniqueAll.append(i[0] + " : " + str(i[1]))
print(uniqueAll)
```

Листинг 4. Фрагмент кода программы
для получения количественных показателей по частям речи

Представим полученные данные в виде графика (рисунок):



Распределение частей речи (в соответствии с библиотекой spaCy)

Синтаксическая характеристика, которую можно извлечь из этих данных, сводится к концентрации в тексте сложноподчиненных предложений (SCONJ [subordinating conjunction] : 1927, т.е. 24 % от общего количества предложений, хотя и с некоторой долей погрешности, поскольку количество подчинительных союзов, определенных spaCy, не обязательно точно совпадет с реальным количеством сложноподчиненных предложений в тексте). Извлечь абсолютно точные сведения о сложносочиненных предложениях статистическим путем на этом этапе исследования не представляется возможным, поскольку определенные в качестве сочинительных союзов токены могут не участвовать в образовании сложносочиненных предложений, а, например, оформлять конъюнкцию второстепенных членов предложения.

Для решения четвертой задачи, направленной на выявление некоторых идиостилистических языковых единиц, по каждой части речи были получены данные о частотности начальных форм (например, для глаголов – форм инфинитива, для существительных – форм единственного числа и т. д.). Начальную форму единицы spaCy определяет автоматически (листинг 5):

```
# Частотность по частям речи
f = open("London/LondonAll01.txt", "r", encoding="utf-8")
text = f.read()
f.close()
doc = nlp(text)
verbs = {}
for token in doc:
   if token.pos_ == "X":
      if token.lemma_ not in verbs.keys():
         verbs[token.lemma_] = 1
      else:
         num = verbs[token.lemma_]
         verbs[token.lemma_] = num+1
```

```
uniqueAll = []
for i in sorted(verbs.items(), key=lambda kv: kv[1], reverse = True):
   uniqueAll.append(i[0] + " : " + str(i[1]))
print(uniqueAll)
```

Листинг 5. Фрагмент кода программы для получения частотных списков по частям речи

Например, среди глаголов лидерами по частотности оказались следующие:

get : 481; go : 414; say : 399; come : 372; see : 305; be : 284; know : 281; have : 251; make : 238; do : 225; take : 177; look : 164; tell : 144; find : 111; want : 96; give : 96; eat : 94; turn : 93; keep : 91; let : 86; answer : 86; start : 85; hear : 84; run : 83; fall : 82; think : 80; break : 79; begin : 77; hold : 76; shake : 71; catch : 71; ask : 66; wait : 66; stand : 63; follow : 62; put : 62; cry : 61; sit : 61; call : 60 и т.д.

Очевидно, что самыми частотными здесь являются самые распространенные английские глаголы, однако уже показательными для исследуемого текста являются *eat* (тема еды – одна из центральных в творчестве Дж. Лондона); *start, run, begin* (характеризуют динамику действий, активность персонажей); *catch* (отождествляется с удачей, получением прибыли); *wait, sit* (характеризуют терпение как добродетель).

Для существительных также приведем первые позиции полученного списка:

smoke : 658; man : 435; time : 217; dog : 201; day : 188; hand : 186; snow : 171; foot : 163; egg : 158; way : 153; eye : 134; dollar : 131; mile : 129; trail : 129; fire : 115; boat : 103; night : 101; hour : 97; ice : 97; head : 95; minute : 93; cabin : 90; camp : 87; face : 86; thing : 83; gold : 82; pound : 79; side : 79; meat : 77 и т.д.

Все существительные составляют своего рода «реферат» к циклу рассказов: *Smoke* – имя главного героя; *man, dog* – основные действующие лица, и тут собаки иногда оказываются важнее и нужнее людей; *day, snow, night, ice* – описание окружающего пространства; *hand, foot, eye, mile, head* – внимание к эмоциям, деталям описания персонажей; *time, day, foot, hour, minute, pound* – подчеркивают важность точности измерений времени, протяженности и веса, внимание к деталям; *way, trail, boat* – характеризуют перемещение на дальние расстояния; *dollar, gold* – тема денег и богатства; *egg, meat* – тема еды.

Приведем также и часть списка прилагательных (точнее, токенов, которые spaCy отнес к прилагательным):

other : 155; good : 132; first : 95; more : 87; big : 85; last : 85; old : 84; same : 72; long : 65; many : 63; own : 62; several : 60; right : 60; next : 56; great : 52; little : 50; young : 50; sure : 47; bad : 44; much : 42; short : 41; white : 41; cold : 41; low : 41; heavy : 40; few : 39; hard : 39 и т.д.

Здесь достаточно интересна первая позиция *other*, что, скорее, является признаком идиостиля писателя. Остальные прилагательные, с одной стороны, являются достаточно распространенными в английском языке, с другой, характеризуют стиль Дж. Лондона как ясный, с четко обозначенными признаками персонажей, предметов и явлений. Заметим, что лидирующие характеристики являются в большинстве своем «положительными» [2].

Таким образом, в ходе представленного исследования нами была разработана процедура нормализации, которую можно применять в отношении практически любых текстов и языков, имеющих буквенное письмо.

Далее нами были написаны программы для получения статистических данных о тексте, например, о количестве токенов и предложений. При этом были использованы возможности библиотеки обработки естественного языка spaCy, что позволило нам также получить достаточно точные данные о синтаксической и частеречной характеристике текста. При этом необходимо учитывать, что под частями речи spaCy понимает достаточно своеобразный набор и наряду с «традиционными» единицами включает в него такие категории как, например, знаки пунктуации и символы. На этом этапе особенностью текста можно считать высокую концентрацию в нем подчинительных союзов.

Интересный материал был получен в результате частотного анализа лексических единиц текста по частям речи, в частности ярко прослеживается важность тем «еда, приемы пищи» и «деньги, богатство», наблюдается частое использование прилагательного *other*. Писатель внимателен к описанию персонажей и пространства.

В качестве перспективы исследования представляется целесообразным проанализировать текст не на уровне отдельных токенов, а обратить внимание на их сочетания, рассматривая паттерны типа «часть речи 1 + часть речи 2» и т.п.; составить размеченный сбалансированный корпус цикла рассказов с целью анализа языковых средств для описания персонажей, пространственных и временных характеристик.

ЛИТЕРАТУРА

1. *Барботько, У. Д.* К вопросу об изучении имен прилагательных (на материале рассказа Дж. Лондона «Костер» на английском и русском языках) / У. Д. Барботько, С. Н. Семенова // Соврем. шк. России. Вопр. модернизации. – 2022. – № 2–1 (39). – С. 116–117.
2. *Горожанов, А. И.* Прикладные аспекты анализа и интерпретации текстов (на материале немецкого и русского языков) / А. И. Горожанов, И. А. Гусейнова. – Казань : О-во с огранич. ответственностью «Бук», 2021. – 208 с.
3. *Девятов, С. Ф.* Развитие главного героя Мартина Идена в романе Джека Лондона / С. Ф. Девятов // Инновации. Наука. Образование. – 2022. – № 50. – С. 2661–2665.
4. *Ефремова, Г. А.* Речевой портрет главного героя : индивидуальные речевые особенности и способы их передачи при переводе (на примере романа Джека Лондона «Мартин Иден») / Г. А. Ефремова, С. А. Истомина // Евраз. Науч. Об-ние. – 2021. – № 6–6 (76). – С. 454–457.

Поступила в редакцию 12.07.2022