Министерство науки и высшего образования Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Филологический факультет

УТВЕРЖДАЮ:
Декан филологического факультета
_____И.В. Тубалова

« 15 » марта 2022 г.

Рабочая программа дисциплины

Язык программирования R

45.03.03 Фундаментальная и прикладная лингвистика

Направленность (профиль) подготовки:
«Фундаментальная и прикладная лингвистика»

Форма обучения
Очная

Квалификация
Бакалавр

Год приема
2021

Код дисциплины в учебном плане: Б1.В.ДВ.06.02

СОГЛАСОВАНО:
Руководитель ОПОП
_____А.В. Васильева

Председатель УМК
_____Ю.А. Тихомирова

Томск – 2022

1. Цель и планируемые результаты освоения дисциплины (модуля)

Целью дисциплины является изучение основных принципов и методов автоматической обработки текстов на естественном языке (ЕЯ)

Целью освоения дисциплины является формирование следующих компетенций:

ПК-4. Способен разрабатывать программный код при решении задач автоматической обработки текстов

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

ИПК-4.2. Создает программный код с использованием языков программирования и манипулирования данными в сфере автоматической обработки текстов

2. Задачи освоения дисциплины

– Изучение методов обработки естественного языка, применение междисциплинарных методов в обработке исследовательских данных.

– Научиться применять понятийный математический аппарат в области лингвистики для решения практических задач профессиональной деятельности.

– Приобрести навыки хранения, структуризации, анализа и визуализации текстового массива данных

3. Место дисциплины (модуля) в структуре образовательной программы

Дисциплина относится к части образовательной программы, формируемой участниками образовательных отношений, предлагается обучающимся на выбор.

4. Семестр(ы) освоения и форма(ы) промежуточной аттестации по дисциплине

Семестр 6, зачет.

5. Входные требования для освоения дисциплины

Для успешного освоения дисциплины требуются компетенции, сформированные в ходе освоения образовательных программ предшествующего уровня образования.

Для успешного освоения дисциплины требуются результаты обучения по следующим дисциплинам: «Введение в языкознание», «Общая фонетика», «Общая морфология», «Общий синтаксис», «Информационные технологии и основы информационной культуры в лингвистике», «Информатика и основы программирования», «Квантитативные методы лингвистики», «Вероятностные модели», «Лингвистические базы данных».

6. Язык реализации

Русский

7. Объем дисциплины (модуля)

Общая трудоемкость дисциплины составляет 2 з.е., 72 часов, из которых:
– лекции: 0 ч.;
– семинарские занятия: 0 ч.
– практические занятия: 32 ч.;
– лабораторные работы: 0 ч.
в том числе практическая подготовка: 32 ч.
Объем самостоятельной работы студента определен учебным планом (38 ч.).

8. Содержание дисциплины (модуля), структурированное по темам

Тема 1. Основы языка программирования R

Синтаксис, объекты, классы, переменные, структуры данных.

Тема 2. Сложные типы данных и работа с ними. Вектор, матрица, массив

Представление структур данных (переменных) в среде R: вектор, матрица, датафрейм, лист.

Тема 3. Управляющие структуры. Условные операторы, циклы, функции

Основные типы управляющих структур, их применение в разных структурах, синтаксис и алгоритм написания

Тема 4. Парсинг и структуризация текстовых данных с помощью языка программирования R

Парсинг web-страниц, основные положения, логика посика и структуризации информации. Библиотека rvest. Извлечение информации через API (Библиотека Rcurl)

Тема 5. Препроцессинг текстовых массивов: токенизация, лемматизация, единый регистр, удалении «шума»

Удаление стоп-слов, лемматизация текстов с помощью стеммеров (mystem)

Тема 6. Словарная поддержка. Типы словарей. Создание словарей

Создание тематических словарей для классификации текстов (sentiment analysis, topic modeling)

Тема 7. Визуализация текстовых данных в R.

Частотный анализ, построение гистограмм,

Тема 8. Автоматический анализ частей речи в библиотеки UdPipe

Принципы разметки, виды и типы морфологических теггеров.

Тема 9. Описательная статистика

Поиск и сравнение лексем в корпусах, метрики сравнения: IPM, TF-IDF, LL-score, коэффициент Жуйана. Базовая статистика, боксплот, тип распределения, корреляции

Тема 10. Разработка чат-бота

Принципы и методы создания час ботов. Разработка чат бота для ПО телеграмм

Тема 11. Итоговая презентация проекта.

9. Текущий контроль по дисциплине

Текущий контроль образовательной программы (темы, раздела, модуля) требованиям образовательных стандартов по направлениям подготовки/специальностям. Текущий контроль успеваемости обучающихся направлен на определение соответствия результатов обучения после освоения элемента по дисциплине проводится путем контроля посещаемости, проведения контрольных работ, тестов по лекционному материалу, разработки кода, выполнения домашних заданий и фиксируется в форме контрольной точки не менее одного раза в семестр. Примерные задания текущего контроля:

Примерные тестовые задания по 2 модулю

#1.Дан вектор целых чисел. Исключить из него a) максимальный b)минимальный элемент.

vec <- c(2,60,4,10)

#2. Дан вектор целых чисел, в котором есть два нулевых элемента. Исключить нулевые элементы.

vec <- c(3,0,7,0,0,3)

#3. Дан вектор X целых чисел и целое число b. Исключить из вектора элементы, равные b.

#4. Дан вектор целых чисел и числа A1, A2 и A3. Включить эти числа в массив, расположив их после второго элемента.

#6. Вывести все элементы вектора, стоящие до максимального элемента

Дан вектор из 20-ти чисел и число A. Вычислить сумму тех отрицательных элементов вектора, значения которых больше, чем A. Подсчитать также количество таких элементов.

#7. Дан вектор из 10-ти чисел. Вычислить среднее арифметическое положительных элементов этого вектора и среднее арифметическое отрицательных элементов этого вектора

#8. Исключить из вектора элементы, расположенные между максимальным и минимальным.

10. Порядок проведения и критерии оценивания промежуточной аттестации

Зачет проводится в письменной и устной форме по выбранному проекту. Проект пердполагает логическое изложение теоретического блока с привязкой к практической деятельности

Итоговый проект представляет собой парсинг, препроцессинг и первичный анализ массива текстов.

1. Составьте словарь слов для поиска в корпусе
2. Определите тип распределения
3. Постройте корреляционный анализ

Измените структуру кода для своих данных:

```
# install.packages("rvest")
library(rvest)
webpage <- read_html("https://news.vtomske.ru/c/tomsk?up=1635835980")
results <- webpage %>% html_nodes(".news-small") %>% html_attr("href")
page_start <- "https://news.vtomske.ru"
# page_full <- paste0(page_start,results[3])
link <- c()
i = 3
while (i <= length(results)) {
  link = append(link, paste0(page_start, results[i]))
  i=i+1
}
webpage <- read_html("https://news.vtomske.ru/c/tomsk?up=1635835980")
pages2 <- webpage %>% html_nodes(".prev") %>% html_attr("href")
page_start <- "https://news.vtomske.ru/c/tomsk"
page_full <- paste0(page_start, pages2)
vec_links <- c()
i <- 1
page_start <- "https://news.vtomske.ru/c/tomsk"
part_pages <- "?down=1637120100"
page_links <- c()
for (i in 1:139) {
  page_full <- paste0(page_start, part_pages)
  webpage <- read_html(page_full)
```

```
  part_pages <- webpage %>% html_nodes(".prev") %>%
    html_attr("href")
  page_links[i] <- paste0(page_start,part_pages)
  #page_links[i] <- page_full
  i=i+1
}
page_links
link
link[1]
scarp_url <- function(url){
  url <- read_html("https://news.vtomske.ru/c/tomsk?up=1637118900")
  results <- url %>% html_nodes(".news-small") %>% html_attr("href")
  return(results)
}
scarp_text <- function(url){
  txt_link = read_html(url)
  text = txt_link %>% html_nodes(".full-text") %>% html_text()
  return(text)
}
scarp_head <- function(url){
  txt_link = read_html(url)
  text = txt_link %>% html_nodes(".material-title") %>% html_text()
  return(text)
}
scarp_date <- function(url){
  txt_link = read_html(url)
  text = txt_link %>% html_nodes(".info") %>% html_text()
  return(text)
}
scarp_text(link[2])
scarp_head(link[2])
scarp_date(link[2])

df <- data.frame(bodyNws = NA,
                 titleNws = NA,
                 dateNws=NA)
i=1
for (i in 1:length(link)) {
  bodyNws = scarp_text(link[i])
  titleNws <- scarp_head(link[i])
  dateNws <- scarp_date(link[i])
  df <- rbind(df, cbind(titleNws,bodyNws,dateNws))
}
df[2,3]
grepl("Сегодня", df[4,3])
Sys.Date()
df$bodyNws[2]
df$bodyNws <- gsub("Дмитрий Кандинский / vtomske.ru",
     "", df$bodyNws)
i=1
for (i in 1:length(df[,3])) {
  grepl("Сегодня", df[i,3])
```

```
  if (grepl("Сегодня", df[i,3])){
    df[i,3] <- gsub("Сегодня",
        Sys.Date(), df[i,3])
  }else if(grepl("Вчера", df[i,3])){
    df[i,3] <- gsub("Вчера",
                Sys.Date()-1, df[i,3])
  }
}

i=1
for (i in 1:length(df$dateNws)) {
  grepl("Сегодня", df$dateNws[i])
  if (grepl("Сегодня", df$dateNws[i])){
    df[i,3] <- gsub("Сегодня",
                Sys.Date(), df$dateNws[i])
  }else if(grepl("Вчера", df$dateNws[i])){
    df[i,3] <- gsub("Вчера",
                Sys.Date()-1, df$dateNws[i])
  }
}


i=1
for (i in 1:length(df$titleNws)) {
  if (grepl("Сегодня", df$dateNws[i])){
    df$dateNws[i] <- gsub("Сегодня", Sys.Date(), df$dateNws[i])

  }
}
i=1
for (i in 1:length(df$titleNws)) {
  if (grepl("Вчера", df$dateNws[i])){
    df$dateNws[i] <- gsub("Вчера", Sys.Date()-1, df$dateNws[i])

  }
}
grepl("Вчера", df$dateNws[5])
gsub("Вчера", Sys.Date()-1, df$dateNws[5])
if ()
Sys.Date()-1


df <- data.frame(bodyNws = NA,
                 titleNws = NA,
                 dateNws=NA)
i=1
link <- c()
page_start <- "https://news.vtomske.ru"
Sys.time()
for (i in 1:length(page_links)) {
 link = append(link, paste0(page_start, scarp_url(page_links[i])))
}
```

```
Sys.time()


i=1
Sys.time()
for (i in 1:length(link)) {
  if (grepl("news.vtomske.ruNA", link[i])==FALSE){
  bodyNws = scarp_text(link[i])
  titleNws <- scarp_head(link[i])
  dateNws <- scarp_date(link[i])
  df <- rbind(df, cbind(bodyNws,titleNws,dateNws))
  }
}
Sys.time()
write.csv(df, "vtomske2021.csv")
```

Результаты зачета определяются оценками «зачтено», «не зачтено».

Критерии зачета обусловлены логической демонстрацией приобретенных компетенций в соответствии с текущей программой. Демонстрация предусматривает уверенное использование терминологии, понимание и корректное использование математического аппарата, предусматривает корректность написания кода, его понимание и корректное использование в нем математических методов. Отметка «зачтено» выставляется за счет демонстрации полученных компетенций в практиках, домашних работах и итоговом задании: уверенное владение и понимание работы кода, знание и демонстрация в практике теоретических основ баз данных. Минимальный порог зачета составляет 55 баллов, ниже 55 – «не зачтено»

11. Учебно-методическое обеспечение

а) Электронный учебный курс по дисциплине в электронном университете «Moodle» - https://moodle.tsu.ru/course/view.php?id=12998

б) Оценочные материалы текущего контроля и промежуточной аттестации по дисциплине.

в) План семинарских / практических занятий по дисциплине.

Семинар №1

1. Основы языка программирования R
2. Переменные, структуры данных
3. Циклы, проверка условий

Семинар №2

1. Сбор и структуризация текстовых данных
2. Работа с файлами
3. Лемматизация текстов при помощи mystem

Семинар №3

1. Библиотека quanteda для векторизации и анализа текстовых массивов данных
2. Векторизация текстов. Принципы, методы
3. Составление словарей, n-грамы

Семинар №4

1. Статистический анализ матрицы слов текста
2. Описательная статистика
3. Корреляционный анализ
4. Проверка статистических гипотез
5. Кластерный анализ

Семинар №6

1. Сокращение пространства признаков

2. Визуализация и анализ текстовых данных

Подготовка к проведению лабораторных работ начинается в начале теоретического изложения изучаемой темы и продолжается по ходу её изучения при освоении материала на занятиях в рамках практических заданий и работе над ним в ходе самостоятельной подготовки дома и в библиотеках. Для качественного выполнения лабораторных работ студентам необходимо:

1) повторить теоретический материал по конспекту и учебникам;

2) ознакомиться с описанием лабораторной работы;

3) в специальной тетради для лабораторных работ записать название и номер работы, перечень необходимого программного обеспечения, подготовить алгоритм или код;

4) выяснить цель работы, четко представить себе поставленную задачу и способы её достижения, продумать ожидаемые результаты опытов;

5) ответить устно или письменно на контрольные вопросы по изучаемой теме или решить ряд задач;

6) изучить порядок выполнения лабораторной работы. Подготовить среду выполнения кода к работе. После проверки правильности алгоритма работы программы преподавателем можно начинать выполнение лабораторной работы.

д) Методические указания по организации самостоятельной работы студентов.

Формы самостоятельной работы студентов разнообразны. Они включают в себя:

– изучение и систематизацию практических и теоретических примеров в рамках выполнения текущих заданий по предмету;

– изучение учебной, научной и методической литературы, материалов периодических изданий с привлечением электронных средств официальной, статистической, периодической и научной информации;

– подготовку докладов и презентаций, написание программного кода и его отладка;

– участие в работе студенческих конференций, комплексных научных исследованиях.

Самостоятельная работа приобщает студентов к научному творчеству, поиску и решению актуальных современных проблем.

Примеры самостоятельной работы студентов:

Создание словаря и иго частотного распределения в текстах:

```
library(readxl)
#library(quanteda.sentiment)
library(quanteda)
# install.packages("remotes")
# remotes::install_github("quanteda/quanteda.sentiment")
tmp <- read_excel("full_word_rating_after_coding.xlsx", col_names = TRUE)

df #head body stem class

mycorp <- corpus(df, text_field = "stem", )



dict <- dictionary(list(negative = c(tmp$word[tmp$value==-1]),
                        neg_negative = c(tmp$word[tmp$value==-2]),
                        pos_positive = c(tmp$word[tmp$value==2]),
                        neutral = c(tmp$word[tmp$value==0]),
                        positive = c(tmp$word[tmp$value==1])))
valence(dict) <- list(negative = -1, neg_negative = -2, pos_positive = 2, neutral = 0, positive
=1)
```

```
dict2 <- dictionary(list(neg = c(tmp$word[tmp$value < 0]),
                         neut = c(tmp$word[tmp$value==0]),
                         pos = c(tmp$word[tmp$value>0])))
valence(dict2) <- list(neg = -1, pos = 1, neut = 0)
polarity(data_dictionary_LSD2015) <- dict
 # list(pos = c("positive", "neg_negative"), neg = c("negative", "neg_positive"))
sent_pres <- mycorp_vk %>%
  corpus_subset(gnd == "f")
sent_pres2 <- mycorp_vk %>%
  corpus_subset(gnd == "m")
summary(mycorp_vk)
x_m <- tokens_lookup(tokens(sent_pres), dictionary = dict2) %>%
  dfm()
x_f <- tokens_lookup(tokens(sent_pres2), dictionary = dict2) %>%
  dfm()
x_m <- dfm_weight(x_m, scheme = "prop")
x_f <- dfm_weight(x_f, scheme = "prop")
x_full <- tokens_lookup(tokens(mycorp_vk), dictionary = dict2) %>%
  dfm()

x_f <-convert(x_f, to = "data.frame")
x_m <-convert(x_m, to = "data.frame")
x_m$gnd <- "f"
x_f$gnd <- "m"
x_full_abs_vkwall <- rbind(x_f, x_m)
x_full_abs_vkwall$ComType <- "vkw"
write.csv(x_full_abs_vkwall, "sentiment_vkwall_gnd.csv")

ggplot(x_full_abs_vkwall, aes(doc_id, neut, fill = gnd, group = gnd)) +
  geom_bar(stat='identity', position = position_dodge(), size = 1) +
scale_fill_brewer(palette = "Set1") +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +
  ggtitle("Sentiment scores in twelve Sherlock Holmes novels") + xlab("")
x_m2 <-  convert(x_m, to = "data.frame")
x_m2 <- as.data.frame(x_m)
x_f2 <-  convert(x_f, to = "data.frame")
x_f2 <- as.data.frame(x_f)
Корреляционный анализ:
library(outliers)
grubbs.test(emot_int$neg, type = 10)

grubbs.test(emot_int$pos, type = 10)
grubbs.test(emot_int$neut, type = 10)
library(ggplot2)
p = ggplot(emot_int[,-1], aes(x=self))
(p <- p+geom_density(aes(fill=gnd), alpha=1/2))

# Sample data
data <- emot_int[, 2:4] # Numerical variables
groups <- as.factor(emot_int[, 5]) # Factor variable (groups)
# Plot correlation matrix
pairs(data)
```

```
# Equivalent with a formula
pairs(~ neg+pos+neut, data = emot_int)


pairs(data,                    # Data frame of variables
      labels = colnames(data),  # Variable names
      pch = 1,                 # Pch symbol
      bg = rainbow(2)[groups],  # Background color of the symbol (pch 21 to 25)
      col = rainbow(2)[groups], # Border color of the symbol
      main = "",    # Title of the plot
      row1attop = TRUE,         # If FALSE, changes the direction of the diagonal
      gap = 1,                  # Distance between subplots
      cex.labels = NULL,        # Size of the diagonal text
      font.labels = 1)          # Font style of the diagonal text

panel.hist <- function(x, ...) {
  usr <- par("usr")
  on.exit(par(usr))
  par(usr = c(usr[1:2], 0, 1.5))
  his <- hist(x, plot = FALSE)
  breaks <- his$breaks
  nB <- length(breaks)
  y <- his$counts
  y <- y/max(y)
  rect(breaks[-nB], 0, breaks[-1], y, col = rgb(0, 1, 1, alpha = 0.5), ...)
  # lines(density(x), col = 2, lwd = 2) # Uncomment to add density lines
}

# Creating the scatter plot matrix
pairs(data,
      upper.panel = NULL,         # Disabling the upper panel
      diag.panel = panel.hist)    # Adding the histograms
# Function to add correlation coefficients
panel.cor <- function(x, y, digits = 2, prefix = "", cex.cor, ...) {
  usr <- par("usr")
  on.exit(par(usr))
  par(usr = c(0, 1, 0, 1))
  Cor <- abs(cor(x, y)) # Remove abs function if desired
  txt <- paste0(prefix, format(c(Cor, 0.123456789), digits = digits)[1])
  if(missing(cex.cor)) {
    cex.cor <- 0.4 / strwidth(txt)
  }
  text(0.5, 0.5, txt,
       cex = 1 + cex.cor * Cor) # Resize the text by level of correlation
}

# Plotting the correlation matrix
pairs(data,
      upper.panel = panel.cor,    # Correlation panel
      lower.panel = panel.smooth) # Smoothed regression lines
```

```
# install.packages("gclus")
library(gclus)

# Correlation in absolute terms
corr <- abs(cor(data))

colors <- dmat.color(corr)
order <- order.single(corr)

cpairs(data,                 # Data frame of variables
     order,                # Order of the variables
     panel.colors = colors,   # Matrix of panel colors
     border.color = "grey70", # Borders color
     gap = 0.45,             # Distance between subplots
     main = "Ordered variables colored by correlation", # Main title
     show.points = TRUE,      # If FALSE, removes all the points
     pch = 21,               # pch symbol
     bg = rainbow(2)[groups]) # Colors by group


#install.packages("psych")
library(psych)

pairs.panels(data,
          smooth = TRUE,      # If TRUE, draws loess smooths
          scale = FALSE,      # If TRUE, scales the correlation text font
          density = TRUE,     # If TRUE, adds density plots and histograms
          ellipses = TRUE,    # If TRUE, draws ellipses
          method = "spearman", # Correlation method (also "spearman" or "kendall")
          pch = 21,           # pch symbol
          lm = FALSE,         # If TRUE, plots linear fit rather than the LOESS (smoothed)
fit
          cor = TRUE,         # If TRUE, reports correlations
          jiggle = FALSE,     # If TRUE, data points are jittered
          factor = 2,         # Jittering factor
          hist.col = 4,       # Histograms color
          stars = TRUE,       # If TRUE, adds significance level with stars
          ci = TRUE)          # If TRUE, adds confidence intervals

library(psych)

corPlot(data, cex = 1.2)

library(ggplot2)
# install.packages("ggExtra")
library(ggExtra)
p_base <- ggplot(emot_int,aes(x=neg,y=pos,color=gnd)) + geom_point()
ggExtra::ggMarginal(p_base, groupColour = TRUE, groupFill = TRUE)
library(otrimle)
clus <- otrimleg(emot_int[,c(1,2)], G=2:5, monitor=1) # параметр monitor позволяет
видеть ход выполнения
```

```
# Equivalent but using the plot function
plot(data)
library(tidyverse)
cor(w_dict_dfm_full[,2:4] %>% w_dict_dfm_mycorp_vk$gnd=="m")
w_dict_dfm_mycorp_vk  %>%
  group_by(gnd) %>%
  plot(w_dict_dfm_mycorp_vk$self)
w_dict_dfm_full$gnd <- as.factor(w_dict_dfm_full$gnd)
w_dict_dfm_full_01 <- w_dict_dfm_full
w_dict_dfm_full_01$gnd01[w_dict_dfm_full$gnd=="m"] <- 1
w_dict_dfm_full_01$gnd01[w_dict_dfm_full$gnd=="f"] <- 0

fitglm <- glm(w_dict_dfm_full_01$gnd01 ~ w_dict_dfm_full_01$personal - w_dict_dfm_full_01$you +
                w_dict_dfm_full_01$we +w_dict_dfm_full_01$self)
```

в) План практических занятий по дисциплине соответствует п. 8.

г) Методические указания по организации самостоятельной работы студентов: самостоятельная работа студентов включает анализ материала по темам с опорой на презентации по темам, подготовку к практическим занятиям.

12. Перечень учебной литературы и ресурсов сети Интернет

а) основная литература:

– Кабаков Р. R в действии. Анализ и визуализация данных на языке R / Роберт И. Кабаков Р., – М.: ДМК, 2016. – 587 с.

– Шипунов А. Б. Наглядная статистика. Используем R. / А. Б. Шипунов, Е. М. Балдин, П. А. Волкова, А .И. Коробейников, С. А. Назарова, С. В. Петров, В. Г. Суфиянов, – М.: ДМК, 2017. – 296 с.

– Мастицкий С.Э. Статистический анализ и визуализация данных с помощью R / Мастицкий С.Э., Шитиков В.К., - М.: ДМК, 2015. – 495с.

б) дополнительная литература:

– Thomas Rahlf. Data Visualisation with R. Springer International Publishing, New York, 2017. ISBN 978-3-319-49750-1.

– Lawrence Leemis. Learning Base R. Lightning Source, 2016. ISBN 978-0-9829174-8-0

– Matthias Kohl. Introduction to statistical data analysis with R. bookboon.com, London, 2015. ISBN 978-87-403-1123-5.

– Torsten Hothorn and Brian S. Everitt. A Handbook of Statistical Analyses Using R. Chapman & Hall/CRC Press, Boca Raton, Florida, USA, 3rd edition, 2014. ISBN 978-1-4822-0458-2.

– An Introduction to Statistical Learning: with Applications in R (Springer Texts in Statistics), Corr. 7th printing / G. James, D. Witten, T. Hastie, R. Tibshirani, Springer, 2017

–JurafskyD., Martin J. Speech and Language Processing.An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition.Prentice Hall, 2000

в) ресурсы сети Интернет:

– открытые онлайн-курсы

– Журнал «Эксперт» - http://www.expert.ru

– Официальный сайт Федеральной службы государственной статистики РФ - www.gsk.ru

– Официальный сайт Всемирного банка - www.worldbank.org

– Общероссийская Сеть КонсультантПлюс Справочная правовая система. http://www.consultant.ru

– Официальный сайт языка программирования R - www.r-cran.com

13. Перечень информационных технологий

а) лицензионное и свободно распространяемое программное обеспечение:

– Microsoft Office Standart 2013 Russian: пакет программ. Включает приложения: MS Office Word, MS Office Excel, MS Office PowerPoint, MS Office On-eNote, MS Office Publisher, MS Outlook, MS Office Web Apps (Word Excel MS PowerPoint Outlook);

– публично доступные облачные технологии (Google Docs, Яндекс диск и т.п.).

– язык программирования R (RStudio) и Python;

– Программа Mystem.

б) информационные справочные системы:

– Электронный каталог Научной библиотеки ТГУ – http://chamo.lib.tsu.ru/search/query?locale=ru&theme=system

– Электронная библиотека (репозиторий) ТГУ – http://vital.lib.tsu.ru/vital/access/manager/Index

– ЭБС Лань – http://e.lanbook.com/

– ЭБС Консультант студента – http://www.studentlibrary.ru/

– Образовательная платформа Юрайт – https://urait.ru/

– ЭБС ZNANIUM.com – https://znanium.com/

– ЭБС IPRbooks – http://www.iprbookshop.ru/

в) профессиональные базы данных *(при наличии)*:

– Университетская информационная система РОССИЯ – https://uisrussia.msu.ru/

– Единая межведомственная информационно-статистическая система (ЕМИСС) – https://www.fedstat.ru/

14. Материально-техническое обеспечение

Аудитории для проведения занятий лекционного типа.

Аудитории для проведения занятий семинарского типа, индивидуальных и групповых консультаций, текущего контроля и промежуточной аттестации.

Помещения для самостоятельной работы, оснащенные компьютерной техникой и доступом к сети Интернет, в электронную информационно-образовательную среду и к информационным справочным системам.

Лаборатории, оборудованные компьютерами (не ниже i3, RAM 8Gb), проектором

Аудитории для проведения занятий лекционного и семинарского типа индивидуальных и групповых консультаций, текущего контроля и промежуточной аттестации в смешенном формате («Актру»).

15. Информация о разработчиках

Степаненко Андрей Александрович, ассистент кафедры общей, компьютерной и когнитивной лингвистики