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

Филологический факультет

УТВЕРЖДАЮ:
Декан
_____И. В.Тубалова
«31» 08 20____ г.

Рабочая программа дисциплины

Информационные технологии в филологии

по направлению подготовки

45.04.01 Филология

Направленность (профиль) подготовки :
Академическая филология: современные исследовательские технологии

Форма обучения
Очная

Квалификация
Магистр

Год приема
2022

Код дисциплины в учебном плане: Б1.В.01

СОГЛАСОВАНО:
Руководитель ОП
_____Т.А. Демешкина

Председатель УМК
_____Ю.А. Тихомирова

Томск – 2022

1. Цель и планируемые результаты освоения дисциплины

Целью освоения дисциплины является формирование следующих компетенций:

УК-4 Способен применять современные коммуникативные технологии, в том числе на иностранном языке, для академического и профессионального взаимодействия;

ПК-1 Выполнение отдельных заданий в рамках решения исследовательских задач в сфере филологии под руководством более квалифицированного работника;

ПК-2 Представление результатов научных исследований в сфере филологии профессиональному сообществу.

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

ИУК-4.1 Обосновывает выбор актуальных коммуникативных технологий (информационные технологии, модерирование, медиация и др.) для обеспечения академического и профессионального взаимодействия.

ИУК-4.2 Применяет современные средства коммуникации для повышения эффективности академического и профессионального взаимодействия, в том числе на иностранном языке.

ИУК-4.3 Оценивает эффективность применения современных коммуникативных технологий в академическом и профессиональном взаимодействиях.

ИПК-1.2 Реализует под руководством более квалифицированного работника план решения исследовательской задачи, используя необходимые информационные ресурсы, методы получения данных, при необходимости производя его корректировку в свете полученных промежуточных результатов.

ИПК-2.2 Использует современные технологии для представления результатов научных исследований в сфере филологии.

2. Задачи освоения дисциплины

– Освоить инструменты ведения отчетности и написания статей.

– Научиться применять основные программные средства с целью увеличения эффективности обработки естественного языка.

– Научиться владеть формальными грамматиками, автоматизировать работу извлечения сущностей из текстового массива данных.

– Уметь интегрировать средства автоматизации обработки естественного языка в программный код.

3. Место дисциплины в структуре образовательной программы

Дисциплина относится к части образовательной программы, формируемой участниками образовательных отношений.

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

Первый семестр, зачет

5. Входные требования для освоения дисциплины

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

6. Язык реализации

Русский

7. Объем дисциплины

Общая трудоемкость дисциплины составляет 2 з.е., 72 часов, из которых:
-лекции: 10 ч.
-практические занятия: 24 ч.
в том числе практическая подготовка: 24 ч.
Объем самостоятельной работы студента определен учебным планом.

8. Содержание дисциплины, структурированное по темам

Тема 1. Интеллектуальные алгоритмы и лингвистика
Информационные технологии в современном мире
1.2. История информационных технологий
1.3. Архитектура и устройство персональных компьютера
1.1. Тенденции развития современных технологий искусственного интеллекта
1.2. Использование интеллектуальных алгоритмов для решения лингвистических задач
1.3. Типы предварительной обработки текста для применения интеллектуальных алгоритмов
1.4. Проблема получения обучающих данных
Тема 2. Примеры интеллектуальных алгоритмов
2.1. Простейшая нейросеть на примере перцептрона
2.2. Обучение модели и оценка качества
2.3. Синонимическая близость и алгебраические операции над векторами
2.4. Рекуррентная модель и кодирование в вектор предложения
2.5. Модели глубокого обучения
Тема 3. Инструменты для разработки интеллектуальных алгоритмов
3.1. Язык Python и технология CUDA
3.2. TensorFlow и репозиторий моделей Google
3.3. Библиотека глубокого обучения PyTorch
3.4. HuggingFace и библиотека Transformers
3.5. Библиотеки NLTK и Pymorphy
Тема 4. Трансдукционные модели и трансформер
4.4. BERT как кодирующая половина трансформера
4.5. Идея Fine tuning
4.6. Генеративные модели и GPT как декодирующая половина трансформера
Тема 5. Организация научных исследований
5.1. Формирование структуры (отчета, диссертации, исследования) и содержания с помощью автоматизации в MS Office
5.2. Автоматическое формирование литературы (Mendeley, Zotero)
Тема 6. Организация научных исследований в Latex
6.1 Введение в Latex
6.2 Организация структуры документа
6.3 Работа с изображениями
6.4 Работа с формулами
6.5 Создание презентаций

9. Текущий контроль по дисциплине

Текущий контроль по дисциплине: контроль посещаемости, проведение контрольных работ (ИУК-4.1, ИУК-4.2, ИУК-4.3), разработка кода (ИПК-1.2), тесты по лекционному материалу (ИУК-4.1, ИПК-2.2). Контроль успеваемости обучающихся направлен на определение соответствия результатов обучения после освоения элемента по дисциплине и фиксируется в форме контрольной точки не менее одного раза в семестр. Примерные задания текущего контроля:

Первоначальный смысл английского слова «компьютер»:
Выберите один ответ:
a. вид АЛУ
b. электронно-лучевая трубка
c. набор ламп, выполняющих различные функции
d. человек, производящий расчеты
e. электронный аппарат
Машины первого поколения были созданы на основе…
Выберите один ответ:
a. электронно-вакуумных ламп
b. транзисторов
c. интегральных микросхем
d. зубчатых колес
Основной элементной базой ЭВМ третьего поколения являются…
Выберите один ответ:
a. микропроцессор
b. интегральные микросхемы
c. электромеханические схемы
d. транзисторы
Основной элементной базой ЭВМ четвертого поколения являются…
Выберите один ответ:
a. электромеханические схемы
b. полупроводники
c. электровакуумные лампы
d. микропроцессор

10. Порядок проведения и критерии оценивания промежуточной аттестации

Зачет по дисциплине принимается на основе достижения рубежных показателей в рейтинге (не ниже 55 баллов), при выполнении практических заданий (ИПК-1.2, ИПК-2.2), тестов (ИУК-4.1, ИУК-4.2, ИУК-4.3), посещения занятий.

Рейтинг, баллы
1 – присутствие на лекции
1 – присутствие на занятии
1-3 – работа на занятии
1-36 – подготовка к занятию и работа на практическом занятии (в т.ч. д/з)

Примерный перечень практических заданий (ИПК-1.2, ИПК-2.2):
Задача:
Генерация заголовков новостей. Скачайте датасет новостей riatomsk.csv. В колонках есть следующие атрибуты: «lead» - лид новости, «title» -заголовок новости, «body» - тело новости. Обучите модель RuGPT-3 с целью генерации заголовка новости. Загрузите полученный код в мудл, прикрепите сгенерированные примеры заголовков в формате .txt.
Пример кода:
%%writefile setup.sh

```
git clone https://github.com/NVIDIA/apex
cd apex
pip install -v --disable-pip-version-check --no-cache-dir ./
!sh setup.sh
import re
import pandas as pd
from sklearn.utils import shuffle

data = pd.read_csv("/content/drive/MyDrive/news.csv",encoding='utf8',index_col=0)

titles1 = data['Head']
print (titles1)

titles = titles1.dropna()
titles.convert_dtypes(convert_string=True)

texts1 = data['Text']
print (texts1)


texts = texts1.dropna()
texts.convert_dtypes(convert_string=True)

# создаем новый датафрейм
data2 = data[["Head", "Text"]]
# удаляем пропуски
data3 = data2.dropna(axis = 0, how = "any")
data4 = data3.astype('string')

titles1 = data['Head']
print (titles1)

titles = titles1.dropna()
titles.convert_dtypes(convert_string=True)

texts1 = data['Text']
print (texts1)


texts = texts1.dropna()
texts.convert_dtypes(convert_string=True)

# создаем новый датафрейм
data2 = data[["Head", "Text"]]
# удаляем пропуски
data3 = data2.dropna(axis = 0, how = "any")
data4 = data3.astype('string')

headlines = data4["Head"]
bodies = data4["Text"]
data5 = pd.concat([headlines, bodies])
```

```
data5 = shuffle(data5)

train = data5
valid = data5[:1000]
valid = shuffle(valid)

import numpy as np
import random
random.seed(1234)
np.random.seed(1234)
val_ind = random.sample(range(data5.shape[0]), 500)

with open("train.txt", "w") as file:
   file.write("\n".join(train))
with open("valid.txt", "w") as file:
   file.write("\n".join(valid))
!python3 pretrain_transformers.py \
   --output_dir=my_model \
   --model_type=gpt2 \
   --model_name_or_path=sberbank-ai/rugpt3small_based_on_gpt2 \
   --do_train \
   --train_data_file=train.txt \
   --do_eval \
   --fp16 \
   --eval_data_file=valid.txt \
   --per_gpu_train_batch_size 1 \
   --gradient_accumulation_steps 1 \
   --num_train_epochs 1 \
   --block_size 1024 \
   --overwrite_output_dir
```

Задача: сгенерировать заголовок новости, дообучив модель ruGPT-3, на основе датасета «riatomsk.csv»
Пример кода для генеративных моделей языка:

```
!pip3 install urllib3==1.26.4
!pip3 install transformers==2.8.0
!pip3 install wget

import wget

wget.download('https://raw.githubusercontent.com/sberbank-ai/ru-gpts/master/generate_transformers.py', './')
wget.download('https://raw.githubusercontent.com/sberbank-ai/ru-gpts/master/pretrain_transformers.py', './')

%%writefile setup.sh
git clone https://github.com/NVIDIA/apex
cd apex
pip install -v --disable-pip-version-check --no-cache-dir ./
!sh setup.sh

import re
import pandas as pd
```

```
from sklearn.utils import shuffle

data = pd.read_csv("/content/drive/MyDrive/news.csv",encoding='utf8',index_col=0)

titles1 = data['Head']
print (titles1)

titles = titles1.dropna()
titles.convert_dtypes(convert_string=True)

texts1 = data['Text']
print (texts1)


texts = texts1.dropna()
texts.convert_dtypes(convert_string=True)

# создаем новый датафрейм
data2 = data[["Head", "Text"]]
# удаляем пропуски
data3 = data2.dropna(axis = 0, how = "any")
data4 = data3.astype('string')

l = pd.Series(data4['Text']).str.replace(r'^.*?\.', '', regex=True)
m = data4['Head']
data4 = pd.concat([m, l], axis=1)

headlines = data4["Head"]
bodies = data4["Text"]
data5 = pd.concat([headlines, bodies])
data5 = shuffle(data5)

train = data5
valid = data5[:1000]
valid = shuffle(valid)

import numpy as np
import random
random.seed(1234)
np.random.seed(1234)

val_ind = random.sample(range(data5.shape[0]), 500)

with open("train.txt", "w") as file:
    file.write("\n".join(train))
with open("valid.txt", "w") as file:
    file.write("\n".join(valid))

!python3 pretrain_transformers.py \
    --output_dir=my_model \
    --model_type=gpt2 \
```

```
--model_name_or_path=sberbank-ai/rugpt3small_based_on_gpt2 \
--do_train \
--train_data_file=train.txt \
--do_eval \
--fp16 \
--eval_data_file=valid.txt \
--per_gpu_train_batch_size 1 \
--gradient_accumulation_steps 1 \
--num_train_epochs 1 \
--block_size 1024 \
--overwrite_output_dir
```

11. Учебно-методическое обеспечение

а) Электронный учебный курс по дисциплине в электронном университете «Moodle» - https://moodle.tsu.ru/course/view.php?id=26935

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

в) План семинарских / практических занятий по дисциплине.

Тема 1. Интеллектуальные алгоритмы и лингвистика. Тенденции развития современных технологий искусственного интеллекта. Использование интеллектуальных алгоритмов для решения лингвистических задач. Основы линейной алгебры. Подготовить презентацию своего исследования

Тема 2. Примеры интеллектуальных алгоритмов. Простейшая нейросеть на примере перцептрона. Извлечение фактов в ПО Tomita-parser

Тема 3. Инструменты для разработки интеллектуальных алгоритмов. Язык Python и технология CUDA. TensorFlow и репозиторий моделей Google.

Тема 4. Трансдукционные модели и трансформер. Решение задач. Простейшая задача машинного перевода на основе модели Seq2seq. Идея долгой краткосрочной памяти. Идея механизма внимания. Обучение модели и оценка качества. Идея замены слов точками в многомерном векторном пространстве. World2Vec, FastText – математические операции над векторами, снятие омонимии

Тема 5. Организация научных исследований. Формирование структуры (отчета, диссертации, исследования) и содержания с помощью автоматизации в MS Office. Автоматическое формирование литературы (Mendeley, Zotero) и интеграция в MS Office

Подготовка к проведению практических 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 поставленной задачи;

3) выяснить цель работы, четко представить себе поставленную задачу и способы её достижения, продумать возможные варианты разработки алгоритмов обработки естественного языка;

5) подготовить среду выполнения кода к работе;

6) писать комментарии в коде, оставлять пометки, уметь отладить программный код, осуществить декомпозицию задачи. 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;

– написание программного кода и его отладка;

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

Примеры самостоятельной работы студентов (математические операции над векторами). В примере поставлена задача построения векторной модели отзывов о товарах, где необходимо найти похожие векторы к качественным продуктам:

```
import numpy as np
import pandas as pd
import re
import nltk
#import spacy
import string
##Чтение датасета с текстом
df = pd.read_csv("reviews.csv", encoding='UTF8', sep="\t")
df.head()
from gensim.models import Word2Vec
from gensim.models.word2vec import LineSentence
w2v_model = Word2Vec(
   min_count=10,
   window=2,
   vector_size=300,
   negative=5,
   alpha=0.03,
   min_alpha=0.0007,
   sample=6e-5,
   sg=1)
#Получаем лист слов
from nltk.tokenize import sent_tokenize
from gensim.utils import simple_preprocess

class MySentences(object):
   def __init__(self, docs):
      self.corpus = docs
   def __iter__(self):
      for doc in self.corpus:
         doc_sentences = sent_tokenize(doc)
         for sent in doc_sentences:
            yield simple_preprocess(sent)
sentences = MySentences(df['text_ready'].tolist())
#Получаем словарь
w2v_model.build_vocab(sentences)
#Обучение
w2v_model.train(sentences, total_examples=w2v_model.corpus_count, epochs=6, report_delay=1)
w2v_model.wv.most_similar(positive=["отлично"], topn=30)
w2v_model.wv.most_similar(positive=["товар", "продукт", "покупка"], topn=50)
```

#Векторы можно складывать и вычитать. Например, рассмотрим такой вариант: “товар” + “продавец” — “ужасный”:

w2v_model.wv.most_similar(positive=["товар", "продавец"], negative=["плохой"], topn=1000)

12. Перечень учебной литературы и ресурсов сети Интернет

а) основная литература:

– Степанов А.Н. Информатика: учебник для вузов / А.Н. Степанов. – СПб.: Питер, 2015 – 720 с.

– Jurafsky Daniel, James H. Martin. Speech and Language Processing. / An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Second Edition. Upper Saddle River, NJ, 2019. https://www.cs.colorado.edu/~martin/slp2.html

– Николаев И.С. / Прикладная и компьютерная лингвистика. Изд. 2 URSS. 2017. 320 с. ISBN 978-5-9710-4633-2

б) дополнительная литература:

– Щипицина Л. Информационные технологии в лингвистике: учеб. пособие / Л. Щипицина. – М.: Флинта, 2015. – 128 с.

– Кодзасов С.В. Алгоритмы преобразования русских орфографических текстов в фонетическую запись / С.В. Кодзасов М.: МГУ, 1970. 130 с/

– Коваль С. А. Лингвистические проблемы компьютерной морфологии. СПб., 2005. Леонтьева Н. Н. Автоматическое понимание текстов. Системы, модели, ресурсы. М., 2006. Ляшевская О. Н. и др. Оценка методов автоматического анализа текста: морфологические парсеры русского языка. Компьютерная лингвистика и интеллектуальные технологии: По материалам ежегодной Международной конференции «Диалог–2010». Вып. 9(16). М., 2010.

– Шаров С. А., Беликов В. И., Копылов Н. Ю., Сорокин А. А., Шаврина Т. О. Корпус с автоматически снятой морфологической неоднозначностью: К методике лингвистических исследований. Компьютерная лингвистика и интеллектуальные технологии. // Диалог. М., 2015. http://www.dialog-21.ru/digests/dialog2015/materials/pdf/SharoffSAetal.pdf

в) ресурсы сети Интернет:

– открытые онлайн-курсы

– морфологический анализатор Mystem https://yandex.ru/dev/mystem/

– язык программирования Python www.python.org

– Извлечение фактов (формальные грамматики) Tomita-parser https://yandex.ru/dev/tomita/

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.).

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

14. Материально-техническое обеспечение

Аудитории для проведения занятий лекционного типа, семинарского типа, индивидуальных и групповых консультаций, в том числе в смешенном формате предполагают использование интерактивнй доски, проектора и системы «Актру»,

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

Практические работы, проведение текущего контроля и промежуточной аттестации подразумевает использование оборудованные аудитории компьютерами (не ниже i3, RAM 8Gb), проектором.

15. Информация о разработчиках

Степаненко Андрей Александрович, старший преподаватель кафедры общей, компьютерной и когнитивной лингвистики Филологического факультета ТГУ.