Федеральное государственное автономное образовательное учреждение высшего образования
"Национальный исследовательский университет
"Высшая школа экономики"

Факультет гуманитарных наук
Школа лингвистики

Рабочая программа дисциплины «Python для гуманитариев - 2»

для цикла свободного минора факультета гуманитарных наук
уровень: бакалавриат

Разработчик программы
Дереза О.В., ассистент, oksana.dereza@gmail.com

Одобрена на заседании школы лингвистики «5» июня 2018 г.
Руководитель школы Е.В. Рахилина _____[подпись]

Рекомендована Академическим советом образовательной программы
«05» июня 2018 г., № протокола 15

Утверждена «5» июня 2018 г.
Академический руководитель образовательной программы
Ю.А. Ландер _____ [подпись]

Москва, 2018

Настоящая программа не может быть использована другими подразделениями университета и другими вузами без разрешения подразделения-разработчика программы.

1 Область применения и нормативные ссылки

Настоящая программа учебной дисциплины из цикла свободного минора устанавливает требования к образовательным результатам и результатам обучения студента и определяет содержание и виды учебных занятий и отчетности.

Программа предназначена для преподавателей, ведущих дисциплину «Python для гуманитариев - 2», учебных ассистентов и студентов, выбравших данный курс.

Программа учебной дисциплины разработана в соответствии с:

- Образовательным стандартом бакалавриата НИУ ВШЭ;
- Программой свободного минора ФГН НИУ ВШЭ.

2 Цели освоения дисциплины

Цель освоения дисциплины «Python для гуманитариев - 2» – научить слушателей применять базовые навыки на языке программирования Python для решения возникающих на практике задач по работе с текстовыми данными: автоматическая обработка и анализ текстовых данных, поиск информации, создание корпусов данных и др.

3 Компетенции обучающегося, формируемые в результате освоения дисциплины

Уровни формирования компетенций:
РБ — ресурсная база, в основном теоретические и предметные основы (знания, умения);
СД – способы деятельности, составляющие практическое ядро данной компетенции;
МЦ – мотивационно-ценностная составляющая, отражает степень осознания ценности компетенции человеком и готовность ее использовать.

В результате освоения дисциплины студент осваивает компетенции:

Компетенция	**Код по ФГОС / ОС ВШЭ**	**Уровень формирования компетенций**
Способен учиться, приобретать новые знания, умения, в том числе в области, отличной от профессиональной	УК-1	МЦ
Способен проводить формализацию лингвистических знаний, анализ и синтез лингвистических структур, квантитативный анализ лингвистических данных с использованием математических знаний и методов	ПК-2	СД, РБ
Способен участвовать в создании представительных текстовых массивов, корпусов текстов, корпусов звучащей речи, мультимодальных корпусов, лингвистических и социолингвистических баз данных и пользоваться этими ресурсами	ПК-11	СД, РБ
Способен проектировать системы анализа и синтеза естественного	ПК-12	СД, РБ

языка, анализа и синтеза мультимодальных языковых систем, в том числе лингвистических компонентов интеллектуальных и информационных электронных систем		
Способен провести квалифицированное тестирование эффективности лингвистически ориентированного программного продукта	ПК-13	СД, РБ
Способен гибко адаптироваться к различным профессиональным ситуациям, проявлять творческий подход, инициативу и настойчивость в достижении целей профессиональной деятельности и личных	ПК-23	МЦ

4 Место дисциплины в структуре образовательной программы

Настоящая дисциплина входит в цикл дисциплин свободного минора факультета гуманитарных наук НИУ ВШЭ. Для освоения учебной дисциплины, студенты должны владеть следующими знаниями и компетенциями:

- Основы работы с языком программирования Python (курс «Python для гуманитариев - 1»);
- основы математики;
- основы лингвистики.

Основные положения дисциплины должны быть использованы в дальнейшем при изучении дисциплин:

- Научно-исследовательский семинар
- Подготовка и защита выпускной квалификационной работы

5 Тематический план учебной дисциплины

№	Название раздела	Всего часов	Аудиторные часы		Самостоя-тельная работа
			Лекции	Семинары	
1	Повторение основ работы с Python		-	2	4
2	Регулярные выражения		2	4	12
3	Основы NLP. Библиотека NLTK.		2	2	10

4	Автоматический морфологический анализ. Mystem.		2	2	14
5	Извлечение именованных сущностей (NER). Библиотеки spacy и natasha.		2	4	14
6	Дистрибутивная семантика. Векторное представление слов. Тематическое моделирование. Библиотека gensim.		2	4	14
7	Классификация и кластеризация		2	2	14
8	**Итого:**	**114**	**12**	**20**	**82**

6 Формы контроля знаний студентов

Тип контроля	Форма контроля	2 или 3 курс				Кафедра/подразделение	Параметры
		1	2	3	4		
	Домашнее задание	*				Школа лингвистики	Три письменных домашних работы, состоящих из нескольких задач

7 Критерии оценки знаний, навыков

- Выполненные домашние задания в формате ipython-тетрадок студенты загружают в свои репозитории на веб-сервисе https://github.com/. Домашние задания, если явно не указано иное, необходимо выложить в репозиторий до 23:59 дня, предшествующего следующему семинару.
- При оценивании программы в первую очередь обращается внимание на то, насколько её работа соответствует требованиям, описанным в задании. Программа, не запускающаяся из-за синтаксических ошибок, не может получить оценку выше 3 баллов. Баллы могут сниматься, в частности, за неточное выполнение задания и отсутствие разбора случаев, из-за которых при исполнении программы может произойти ошибка. Во вторую очередь могут оцениваться оптимальность решения (в смысле времени работы программы и количества строк кода) и стиль.
- Каждая домашняя работа состоит из 3-5 задач разного уровня сложности. Для получения положительной оценки необходимо решить задачи, написав программу на языке Python.
- При обнаружении плагиата в домашнем или контрольном задании это задание получает оценку 0 баллов.

8 Содержание дисциплины

1. **Повторение основ работы с Python**
   Переменные и функции. Типы и структуры данных. Условные выражения. Циклы. Работа с файлами.
2. **Регулярные выражения**

Язык регулярных выражений. Работа с регулярными выражениями в Python. Библиотека re.

3. **Основы NLP.**
   Основы компьютерной обработки текстовых данных. Частотные списки, стоп-слова, закон Ципфа. N-граммы. Корпуса текстов.
4. **NLTK**
   Библиотека для автоматической обработки текстовых данных NLTK. Токенизация, сплиттиннг, стемминг, автоматический морфологический анализ.
5. **Автоматический морфологический анализ.**
   Лемматизация. Частеречная раметка. Морфологический анализатор Mystem. Библиотеки pymystem3 и pymorphy2. Библиотека spacy.
6. **NER**
   Задача извлечения именованных сущностей. Библиотеки spacy и natasha.
7. **Классификация и кластеризация**
   «Мешок слов». Классификация и кластеризация текстов.
8. **Дистрибутивная семантика**
   Дистрибутивная семантика. Векторное представление слов. Тематическое моделирование. Библиотека gensim.

9 Образовательные технологии

Для изучения дисциплины необходим компьютер и следующее программное обеспечение: текстовый редактор Notepad++ или любой другой, поддерживающий подсветку синтаксиса, переключение между разными кодировками и поиск с использованием регулярных выражений; интерпретатор языка Python и изучаемые в рамках курса библиотеки.

Рекомендуемые образовательные технологии включают лекции, практические занятия, самостоятельную работу студентов (выполнение практических домашних заданий с использованием специализированного компьютерного инструментария).

10 Оценочные средства для текущего контроля и аттестации студента

10.1 Оценочные средства для оценки качества освоения дисциплины в ходе текущего контроля

1. Открыть файл "literary_anecdotes.txt". Посчитать,сколько раз упоминается фамилия "Гоголь" в всех формах. Заменить всех Достоевских на Толстых, а Толстых -- на Достоевских, т.е. после замены Лев Толстой во всех формах должен стать Львом Достоевским, а Фёдор Достоевский -- Фёдором Толстым.

2.

а) Скачать текст "Капитанской дочки" или любого другого большого произведения на русском языке, сохранить в файле формата .txt и затем в программе прочитать текст из файла. Очистить текст от пунктуации. Обратите внимание, что типичная для русского языка пунктуация вроде кавычек-ёлочек и длинных тире должна быть удалена, а дефисы между словами должны остаться. Подсказка: первое действие -- удалить всю "приклеенную" к словам пункуацию, второе действие -- удалить тире.

б) Лемматизировать все слова в тексте с помощью pymystem3.

в) Создать частотный список лемм в тексте. Совпадают ли первые 50-100 слов в нем с верхушкой частотного словаря НКРЯ в целом? А если посмотреть подкорпус художественной литературы?

г) Избавиться от стоп-слов и снова посмотреть частотный список. Напечатать 100 самых частотных слов (слово и его частота). Говорят ли они нам что-нибудь о тексте? Можем ли мы определить тему текста, не читая его? А стиль? Или что-нибудь еще?

д) Разбить текст на биграммы, создать частотный список биграмм, напечатать первые 100. Дают ли они нам какую-нибудь дополнительную информацию по сравнению с униграммами? Заметили ли вы в них что-нибудь странное? Почему так получилось?

е) Посчитать количество слов (униграмм) в тексте. Посчитать, сколько слов имеют частоту < 6 и какой это процент от общего числа слов. Напечатать 10 слов с частотой 1, 10 слов с частотой 2 -- и так до частоты 5, а затем выбросить все низкочастотные слова (с частотой меньше 6). Сохранить получившийся после всех этих манипуляций (лемматизация + удаление стоп-слов + удаление низкочастотного хвоста) в новый файл.

3.

а) Взять какой-нибудь небольшой текст (например, рассказ или длинное стихотворение), сохранить его в txt-файл.

б) Загрузив текст из этого файла, сделать полный грамматический разбор и записать результаты в файл json.

в) Написать регулярное выражение, которое будет извлекать из тега только часть речи.

г) Пройтись циклом по списку с разборами, который выдал pymystem3, извлекая из каждого разбора форму слова и его часть речи и записывая их в новый словарь (форма -- ключ, часть речи -- значение).

д) Посчитать абсолютную частоту для всех частей речи, а зате относительнную частоту (абсолютная частота / длина текста).

11 Порядок формирования оценок по дисциплине

Накопленная оценка за домашние задания равна среднему арифметическому оценок за все домашние задания. Итоговая оценка выставляется по накопленной. Способ округления: арифметический.

$$О_{нак} = (О_{дз1} + О_{дз2} + О_{дз3}) / 3$$

$$О_{итог} = О_{нак}$$

12 Учебно-методическое и информационное обеспечение дисциплины

12.1 Базовый учебник

Курс лекций.

12.2 Основная литература

Джеффри Фридл. Регулярные выражения (3-е издание). Символ-плюс: М., 2008 (главы из книги).

Марк Лутц. Изучаем Питон (4-е издание). Символ-плюс: М., 2011

12.3 Интернет-ресурсы

- Документация по языку Python: http://docs.python.org/
- Steven Bird, Ewan Klein, Edward Loper. Natural Language Processing with Python: http://www.nltk.org/
- Документация по Mystem: https://tech.yandex.ru/mystem/doc/index-docpage/
- Симулятор консоли Linux: http://bellard.org/jslinux/

12.4 Программные средства

- Текстовый редактор Notepad++ или любой другой, поддерживающий подсветку синтаксиса, переключение между разными кодировками и поиск с использованием регулярных выражений;
- Интерпретатор языка Python (http://www.python.org/download/ или https://www.anaconda.com/download/).

13 Материально-техническое обеспечение дисциплины

Практические занятия проводятся с использованием мультимедийного проектора в компьютерных классах.