

Министерство образования и науки Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»

Кафедра математической
кибернетики и компьютерных наук

**РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ РЕШЕНИЯ
МАТЕМАТИЧЕСКИХ ЗАДАЧ**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

Студентки 4 курса 411 группы
направления 02.03.02 — Фундаментальная информатика и информационные
технологии
факультета КНиИТ
Селиной Ирины Алексеевны

Научный руководитель
доцент, к. ф.-м. н.

А. С. Иванова

Заведующий кафедрой
к. ф.-м. н.

С. В. Миронов

Саратов 2018

СОДЕРЖАНИЕ

| | |
|--|----|
| ВВЕДЕНИЕ | 3 |
| 1 Основное содержание работы | 4 |
| ЗАКЛЮЧЕНИЕ | 12 |
| СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ | 13 |

ВВЕДЕНИЕ

Нередко при решении всякого рода математических задач возникает потребность в получении ответа в числовом виде. Особенно подобная ситуация характерна при построении графиков, необходимых для наглядности полученных результатов. При этом следует учитывать, что на практике при решении многих задач отсутствует окончательная формула, представляющая ее решение, но известно, что в принципе она существует. Более того, даже если при аналитическом решении удалось вывести подобную формулу, это не будет гарантировать ее продуктивного применения для нахождения конкретных значений решения. В вышеперечисленных случаях зачастую применяются вычислительные, то есть численные, методы. Таким образом, использование численных методов решения математических задач остается важным и необходимым в разных областях науки и практики и по сей день.

Целью выпускной квалификационной работы является создание веб-приложения, позволяющего решать различные математические выражения численными методами. Для достижения данной цели были поставлены следующие задачи:

1. выбор технологии для создания веб-приложения;
2. разработка дизайна приложения и создание элементов интерфейса;
3. разработка функционала, а именно интеграция численных методов и символьных вычислений, построение графиков и написание анализатора формул.

Бакалаврская работа содержит следующие разделы:

- введение;
- описание используемых методов и технологий;
- разработка приложения;
- заключение.

1 Основное содержание работы

Описание используемых методов и технологий. В данной главе были описаны основные понятия, используемые в работе, численные методы решения математических задач, технологии разработки веб-приложений и основы машинного обучения.

В первой части данной главы были рассмотрены численные методы решения обыкновенных дифференциальных уравнений, дифференциальных уравнений в частных производных и вычисления интегралов.

Как известно, чтобы вычислить корни дифференциального уравнения, требуются значение зависимой переменной и значения производных при определенных значениях независимой переменной. Существуют следующие виды задач дифференциальных уравнений — это краевая задача и задача Коши. Согласно источнику [1] решаемая задача является краевой, если дополнительные условия указаны для двух или более значений неизвестной переменной. Если же условия указаны только для одного значения неизвестной переменной — то речь идет о задаче Коши. Результат решения такой задачи для дифференциального уравнения — это некая таблица значений, поскольку подлежащая нахождению функция находится дискретно с определенным шагом, который задается в условии задачи. Для получения этого решения были рассмотрены такие численные методы как Эйлера, Гюна и Рунге—Кутта.

В качестве дифференциальных уравнений в частных производных были рассмотрены такие классические уравнения математической физики, как уравнение теплопроводности, уравнение распространения колебаний и уравнение Пуассона. В качестве численного метода решения таких уравнений был рассмотрен метод сеток, или как его еще называют метод разностей, который является одним из часто используемых численных методов в решении задач, связанных с использованием дифференциальных уравнений в частных производных. Согласно источнику [2] при решении таких задач является обязательным указание начальных и граничных условий. В методе для того, чтобы рассчитать шаги по пространству Δ и по времени τ , необходимо покрыть область расчета (x, t) сеткой, состоящей из $N \times M$ точек. Таким образом фиксируются точки для самого нахождения решения. Следующим шагом необходимо провести замену дифференциальных уравнений в частных производных на уравнения в конечных разностях. При этом для каждой n -

ой точки получившейся сетки вычисляются разностные уравнения. Прделав вышеизложенные действия, получают разностную схему. Разностные схемы бывают двух видов: явные и неявные. Явной называют схему, в которой неизвестное значение искомой функции на $(i + 1)$ -м шаге стоит только в левой части, а в правой части стоят уже вычисленные ранее значения функции. Если в правой части стоят неизвестные значения функций, то схема — неявная. В данной главе были рассмотрены пути построения различных разностных схем для вышперечисленных уравнений классической математической физики.

При аналитическом решении задач интегрированием могут возникнуть трудности, которые могут быть вызваны сложностью расчетов или же невозможностью поиска первообразной функции из-за отсутствия приемлемых выражений. В таких случаях применяют численные методы, самые известные из которых были рассмотрены в данной главе. Для численного вычисления интегралов необходимо заменить функцию, стоящую под знаком интеграла, на ее приближение, для чего требуется провести некоторое заданное число вычислений с определенной точностью. Были рассмотрены методы прямоугольников, трапеций, Симпсона и Монте-Карло.

Во второй части главы были рассмотрены основные технологии создания веб-приложений на стороне сервера и на стороне клиента, которые также представлены в источниках [3]- [4]. На сегодняшний день разработано множество языков программирования с разным уровнем сложности, различными возможностями, собственными преимуществами и недостатками. Многие из них предоставляют возможность реализации веб-приложений. Ниже указан список технологий веб-программирования (одни из самых популярных на сегодняшний день), которые используются на стороне сервера:

- ASP;
- ASP.net;
- Java;
- Perl;
- PHP;
- Python;
- Nodejs.

В свою очередь клиентскую часть реализуют следующие технологии:

- Разработка пользовательского интерфейса:
 - HTML, XHTML;
 - CSS
- Создание и обработка запросов, создание независимого от браузера интерфейса:
 - ActiveX;
 - Adobe Flash, Adobe Flex;
 - Java;
 - JavaScript;
 - Silverlight.

Также были перечислены популярнейшие фреймворки для создания веб-приложений, некоторые из них были рассмотрены более подробно.

В третьей части данной главы были рассмотрены основные понятия машинного обучения и области его применения, такие как ранжирования веб-страниц, перевод документов, контроль доступа, распознавание объектов и так далее.

Также были рассмотрены искусственные нейронные сети, являющиеся продуктом машинного обучения, который, при правильном проектировании, обрабатывает данные, изучает динамику и впоследствии обеспечивает информативный результат. Согласно источнику [5] архитектура искусственной нейронной сети определяет, как ее нейроны расположены или размещены относительно друг друга. Их расположение структурировано для направления синаптических связей нейронов. Топология данной нейронной сети в рамках конкретной архитектуры может описывать различные структуры, которые она может принять. Одной из наиболее важных особенностей искусственных нейронных сетей является их способность учиться из представления образцов (паттернов), которые выражают поведение системы. Следовательно, после того, как сеть узнала о взаимосвязи между входами и выходами, она может обобщить решения, а это означает, что сеть может производить вывод, который близок к ожидаемому (или желаемому) выводу любых заданных входных значений. Согласно источнику [6] одной из часто используемых архитектур искусственных нейронных сетей является сверточная нейронная сеть. В основном они применяются в распознавания изображений. Ее работу можно охарактеризовать, как переход от конкретных особенностей изображе-

ния к более абстрактным деталям, и далее к еще более абстрактным деталям вплоть до выделения понятий высокого уровня. При это сеть самонастраивается и сама определяет необходимую иерархию абстрактных признаков, фильтруя маловажные детали и выделяя существенные.

Разработка приложения. В ходе написания дипломной работы было разработано веб-приложение для решения различных математических задач. Для разработки был выбран фреймворк Django, поскольку он предоставляет все необходимые средства для удобной и быстрой разработки.

На стартовой странице пользователю предоставляется возможность решить какое-либо уравнение и перейти на другие страницы для решения дифференциальных или же интегральных уравнений. Также на ней представлены основные обозначения для ввода выражений и примеры их использования.

Для отображения страницы и решения уравнений используется представление Calc, которое наследуется от класса ExpressionView. Класс ExpressionView отвечает за выполнение действий, которые являются идентичными для нескольких страниц. Таким действием является, например, взятие выражения из формы ввода.

Данные, введенные в форму ввода выражения, обрабатываются анализатором parse. Действия анализатора основываются на применении регулярных выражений. Анализатор включает в себя следующие компоненты:

- список, состоящий из операторов и операций. Данный список необходим для поиска всех переменных, находящихся в заданном выражении;
- список правил, по которому входная формула преобразуется в заданный правилом вид;
- функция `_is_digit(character)`. Данная функция необходима для проверки, является ли рассматриваемый символ цифрой, что требуется для того, чтобы убрать из входного выражения все цифры на этапе выделения переменных;
- функция `parse(expression)`. Она преобразует выражение, на выходе функции получаем измененное выражение и список используемых в нем переменных, который необходим для удобства дальнейших вычислений;
- функция `_apply_rules(expression)`. Данная функция применяет имеющиеся правила для полного преобразования поданного на вход выражения.

После распознавания формулы полученные данные обрабатываются спе-

циальной функцией `parse_expr(expression)` из модуля `sympy` и затем используется функция `solve(expression)` для решения. Для сохранения графика полученного выражения вызывается функция `make_image(expressions, file_name)`.

Таким образом, данные, введенные пользователем, обрабатываются классом `Calc` и затем отрисовываются на `html`-странице `single.html`, форма которой задана классом `ExpressionForm`.

Для решения обыкновенного дифференциального уравнения пользователю необходимо выбрать один из методов, ввести интервал, начальное значение y и число разбиений. Для решения предоставляется три метода: Эйлера, Гюна и Рунге—Кутта, для применения которых были написаны соответствующие функции. Введенные пользователем некоторые данные обрабатываются представлением `CalcODE(ExpressionView)`, затем полученные данные отрисовываются на `html`-странице `ode.html`, форма которой задана классом `ODEForm`. Кроме численного решения пользователю предоставляется общее решение введенного уравнения, частное решение, формулы для расчета выбранным численным методом и два графика, представленные на одной координатной плоскости (первый график построен по точкам, которые были получены численными методами; второй график построен по общему решению).

На странице для решения классических уравнений математической физики пользователь может выбрать уравнение колебаний струны или же уравнение теплопроводности. Если выбрано уравнение теплопроводности, то пользователю необходимо определить значения определенных параметров, таких как функция $f(x, t)$, a , l и так далее. В частности, параметр σ играет немаловажную роль, поскольку от его значения зависит выбор разностной схемы решения. Для решения предоставляется три схемы: явная, неявная и неявно-явная. Введенные пользователем некоторые данные обрабатываются представлением `ThermalPDEView(PDEMethodView)`, затем полученные данные отрисовываются на `html`-странице `thermal_pde.html`, форма которой задана классом `ThermalPDEForm`. Кроме численного решения пользователю предоставляется шаблон схемы, используемой для решения, а также трехмерный график, построенный по получившимся точкам.

Если выбрано уравнение колебаний струны, то пользователю так же необходимо определить значения определенных параметров, таких как функция $f(x, t)$, a , и так далее. В частности, от выбора значения параметра σ зави-

сит выбор разностной схемы решения. Для решения предоставляется три схемы: явная, неявная и неявно-явная. Введенные пользователем некоторые данные обрабатываются представлением `StringPDEView(PDEMethodView)`. Кроме численного решения пользователю предоставляется шаблон схемы, используемой для решения, а также трехмерный график, построенный по полученным точкам.

При выборе численных методов для вычисления интегралов пользователю необходимо выбрать один из методов, ввести интервал и число разбиений. Для решения предоставляется четыре метода: прямоугольников, трапеций, Симпсона и Монте—Карло. Для применения данных методов были написаны соответствующие функции. Введенные пользователем некоторые данные обрабатываются представлением `CalcIntegral(ExpressionView)`, затем полученные данные отрисовываются на html-странице `integral.html`, форма которой задана классом `IntegralForm`. В результате пользователю предоставляется значение определенного интеграла, неопределенного интеграла, посчитанного численным методом и аналитически, формулы для расчета выбранным численным методом и график, характеризующий этот метод.

При выборе численных методов для вычисления двойных интегралов пользователю так же необходимо выбрать один из методов, ввести интервалы и число разбиений. Для решения предоставляется два метода: прямоугольников и Монте—Карло. Для применения данных методов были написаны соответствующие функции. Введенные пользователем некоторые данные обрабатываются представлением `CalcDoubleIntegral(ExpressionView)`, после чего полученные данные будут отрисовываться на html-странице, а именно на странице `double_integral.html`, форма этой страницы задается классом `DoubleIntegralForm`. Кроме численного решения пользователю предоставляется значение определенного интеграла, неопределенного интеграла, формулы для расчета выбранным численным методом и график, характеризующий этот метод.

Кроме всего этого, пользователям предоставляется возможность загрузить изображение с математическим выражением для дальнейшего его решения. Для этого была обучена нейронная сеть.

Для распознавания математических выражений требуется набор изображений, на которых представлены цифры, буквы латинского алфавита, математические символы. Для генерации данных была написана отдельная про-

грамма, компонентами которой являются:

- список, содержащий сами символы;
- список, содержащий представление каждого символа;
- `convert_image(image)` — функция для нормализации изображений;
- `generate_data()` — функция для генерации изображений.

Генерация данных происходит следующим образом. С помощью библиотеки PIL изображаются все символы из представленного списка. Для начертания используются все стандартные шрифты Windows. К таким шрифтам относятся, например, Arial, Condara, Gabriola, Malgun и так далее.

Все изображения нормализуются. Благодаря функции нормализации изображение обрезается так, чтобы символ оказался точно в центре, а также происходит изменение размера изображения до необходимого.

Для самого обучения нейронной сети используется библиотека Keras, а также TensorFlow в качестве вычислительного бэкенда.

Использоваться будет сверточная нейронная сеть, которая в свою очередь состоит из следующих слоев:

1. Первый скрытый слой представляет собой сверточный слой, называемый Conv2D. Слой свёртки умножает значения ядра свертки на исходные значения пикселей изображения (производится поэлементное умножение), после чего все эти умножения суммируются. То есть, каждая уникальная позиция введенного изображения производит число. Слой содержит 50 карты признаков, которые имеют размер 5×5 . Используемая функция активации — `relu`.
2. Слой пулинга, который также называется слоем подвыборки. Размер пула 2×2 . Слой пулинга представляет собой нелинейное уплотнение карты признаков, при этом группа пикселей уплотняется до одного пикселя, проходя нелинейное преобразование. При этом использовалась функция максимума. Преобразования затрагивают непересекающиеся квадраты, каждый из которых ужимается в один пиксель, при этом выбирается пиксель, имеющий максимальное значение.
3. Слой Dropout — отбрасывает (обнуляет) выходы некоторых нейронов, выбираемых случайно и заново для каждого обучающего примера. В данном случае отбрасывается 20%.
4. Слой Flatten — для преобразования данных.

5. Полносвязный слой из 128 нейронов с функцией активации `relu`.
6. Выходной слой, содержащий 38 классов и функцию `softmax` для вывода вероятности для каждого класса.

Для распознавания выражения с изображения также была написана отдельная функция `recognize(img_path)`. Она необходима для выполнения следующих действий: на изображении ищутся все контуры объектов и вокруг найденных контуров выделяется прямоугольная область. Предполагается, что в прямоугольной области содержится только один объект. Прямоугольная область нормализуется, то есть изменяются размеры. Полученная область изображения подается на вход сформированной в результате обучения нейронной сети модели. Для написания функции использовалась библиотека `OpenCV`.

ЗАКЛЮЧЕНИЕ

В ходе написания выпускной квалификационной работы были рассмотрены различные численные методы для решения уравнений, технологии создания веб-приложений, возможности машинного обучения. В результате было создано веб-приложение, позволяющее реализовывать различные расчеты. Были интегрированы численные методы для решения дифференциальных уравнений методами Эйлера, Гюна, Рунге—Кутта, численные методы для решения классических уравнений математической физики, а именно для уравнения теплопроводности и уравнения колебания струны. Кроме численного решения приложение предоставляет общий ответ, а также строится график. Были добавлены методы численного интегрирования. Созданное приложение включает в себя функцию распознавания математических уравнений с изображений, что значительно может упростить работу, если выражение имеет большую длину. Для разработки был выбран фреймворк Django, поскольку он предоставляет все необходимые средства для удобной и быстрой разработки.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Методы численного дифференцирования и интегрирования [Электронный ресурс]. — URL: <http://mathhelpplanet.com/static.php?p=metody-chislennogo-differentsirovaniya-i-integrirvaniya> (Дата обращения 27.05.2018). Загл. с экр. Яз. рус.
- 2 Методы решения задач математической физики [Электронный ресурс]. — URL: <http://pandia.ru/text/80/448/27640.php> (Дата обращения 29.05.2018). Загл. с экр. Яз. рус.
- 3 Server side web frameworks [Электронный ресурс]. — URL: https://developer.mozilla.org/en-US/docs/Learn/Server-side/First_steps/Web_frameworks (Дата обращения 19.05.2018). Загл. с экр. Яз. англ.
- 4 Архитектура web-приложений [Электронный ресурс]. — URL: https://studref.com/384482/informatika/arhitektura_prilozheniy (Дата обращения 19.05.2018). Загл. с экр. Яз. рус.
- 5 Искусственные нейронные сети [Электронный ресурс]. — URL: <https://github.com/wisestump/ComputerVision-Captcha/blob/master/documentation/report.md> (Дата обращения 18.05.2018). Загл. с экр. Яз. рус.
- 6 Сверточные нейронные сети [Электронный ресурс]. — URL: <http://ru.datasides.com/code/cnn-convolutional-neural-networks/> (Дата обращения 20.05.2018). Загл. с экр. Яз. рус.