

# Разработка системы создания курсов адаптивного обучения

Д.А. Добряков, группа 16.Б10-мм

Научный руководитель: проф. каф. СП, д.ф.-м.н., А.Н. Терехов

Научный консультант: ст. преп. каф. СП, М.Х. Немешев

---

Здравствуйте, меня зовут Дмитрий Добряков, и тема моей работы "Разработка системы создания курсов адаптивного обучения".

# Адаптивное обучение

- Подстраивается под обучающегося
- Учитывает индивидуальные особенности
- Обновляемая оценка характеристик
- Снимает риски неосознанной некомпетенции

В первую очередь нужно разобраться с тем, что такое адаптивное обучение. Адаптивное обучение, в отличие от традиционного, направлено на одного конкретного обучающегося. Оно должно учитывать его текущие знания, цели обучения и прочие индивидуальные особенности, а также поддерживать обновляемую оценку всех этих характеристик, для того чтобы корректировать план обучения и наиболее оптимально соответствовать студенту.

Адаптивное обучение повышает эффективность и скорость обучения, обеспечивает более гибкий учебный процесс и позволяет снять риски наличия неосознанной некомпетенции. Тема особенно актуальна в настоящее время, когда многие компании и образовательные учреждения переходят на дистанционный режим, и растет популярность онлайн-образования.

# Постановка задачи

Разработать систему адаптивного обучения, позволяющую создавать и проходить адаптивные курсы:

- Исследовать существующие разработки и алгоритмы в области адаптивного обучения
- Разработать архитектуру приложения
- Подобрать и реализовать алгоритмы поиска образовательной траектории и адаптивного тестирования
- Реализовать систему адаптивного обучения с использованием выбранных алгоритмов
- Построить при помощи системы адаптивный курс и проанализировать результаты его прохождения реальными людьми

3 / 14

Целью работы является разработка системы адаптивного обучения, позволяющей создавать и проходить адаптивные курсы, и для реализации этой цели были поставлены следующие задачи:

- Исследовать существующие разработки и алгоритмы в области адаптивного обучения
- Разработать архитектуру приложения
- Подобрать и реализовать алгоритмы поиска образовательной траектории и адаптивного тестирования
- Реализовать систему адаптивного обучения с использованием выбранных алгоритмов
- Построить при помощи системы адаптивный курс и проанализировать результаты его прохождения реальными людьми

# Обзор: Адаптивный курс

- Модель курса
  - Модули обучения
  - Адаптивная разметка
- Модель студента
  - Прогресс, история
  - Характеристики
- Алгоритмическая модель
  - Индивидуальная траектория
  - Адаптивное тестирование

4 / 14

Что нужно для построения адаптивного курса? Адаптивный курс можно поделить на нескольких частей:

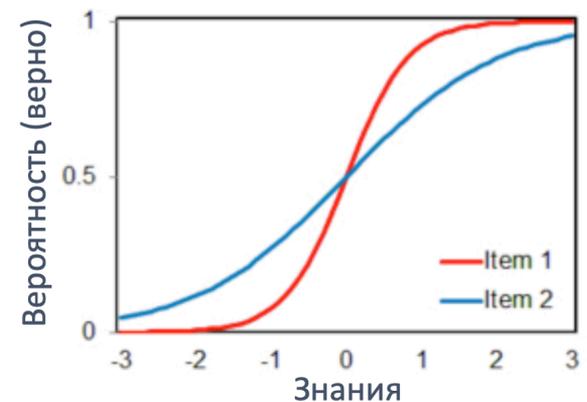
- Во-первых, это модель курса, содержащая в себе его структуру, материалы уроков, задания для проверки этих материалов и различную мета-информацию для дальнейшего обеспечения адаптивности;
- Также, это модель студента, которая содержит данные о студенте, проходящем курс: его прогрессе и степени усвоения материала, индивидуальных особенностях и так далее;
- И наконец, это алгоритмическая модель, которая берет данные из первых двух моделей, и складывает их в алгоритмы, позволяющие эффективно построить ход курса и направить его на данного конкретного студента.

Основной задачей алгоритмической модели является поиск индивидуальной траектории, то есть последовательности образовательных модулей, наиболее подходящей конкретному обучающемуся.

Для решения этой задачи немаловажным является адаптивное тестирование студента, проходящего курс, для определения степени усвоения им материала и дальнейшего использования полученных оценок при построении траектории.

# Обзор: Адаптивное тестирование

- Количественная оценка знаний студента
- Динамическое формирование заданий
- Item Response Theory
  - Модель ответа
  - Резерв тестов
  - Item Characteristic Curve



Конечная цель адаптивного тестирования – определить количественную оценку знаний студента. Адаптивное тестирование характеризуется динамическим формированием заданий, основывающимся на предыдущих ответах.

Для обеспечения адаптивного тестирования часто применяется подход Item Response Theory, согласно которому для каждого задания необходимо иметь некоторую модель ответа, описывающую как студенты отвечают на вопрос в зависимости от их уровня знаний. Также необходимо иметь некоторый резерв заданий, из которого они будут динамически выбираться наиболее оптимальным образом.

Item Response Theory основывается на вероятностном моделировании результатов тестирования. Так, вероятность того, что студент с фиксированным заданным уровнем знаний правильно ответит на вопрос, определяется кривой ICC. Эта кривая, показанная на рисунке, может определяться по-разному, и калибруется для каждого задания.

# Существующие разработки

- Area9 Rhapsode, Knewton
- Компетенции: ФДО ТУСУР
- Эвристические алгоритмы:
  - Генетический
  - Метод роя частиц
  - Прочие

В настоящее время исследования в области адаптивного обучения очень активно проводятся, и были предложены разные подходы к построению индивидуальной траектории, основанные на приближенных методах, таких как генетический алгоритм, метод роя частиц и другие.

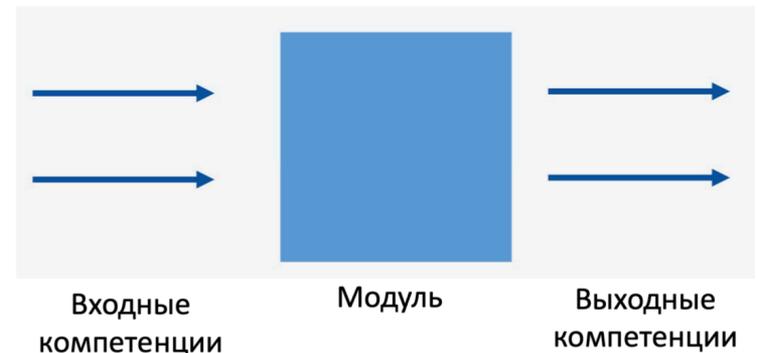
Так, например, исследователи из Томского государственного университета систем управления описали модель курса, основанную на понятии компетенций, и использовали генетический алгоритм для генерации траектории.

К готовым системам можно отнести, например, Area9 Rhapsode, где обучение происходит за счет ответов на вопросы с указанием уверенности в ответе. Система ориентирована на повышение профессиональной квалификации работников, и разработкой курса на заказ занимается специальная команда.

Также есть Knewton – первая в мире система адаптивного обучения, которая в качестве эксперимента уже успешно внедрялась в процесс образования и по итогам улучшила результаты студентов на приблизительно 20%.

# Реализация: Структура данных

- Модули
  - Входные/выходные компетенции
  - Сложность, длительность
- Компетенции
  - Покрываются тестами
- Тесты
  - Сложность
- Студент
  - Уровень компетенций
  - История модулей
  - Адаптированная сложность модулей



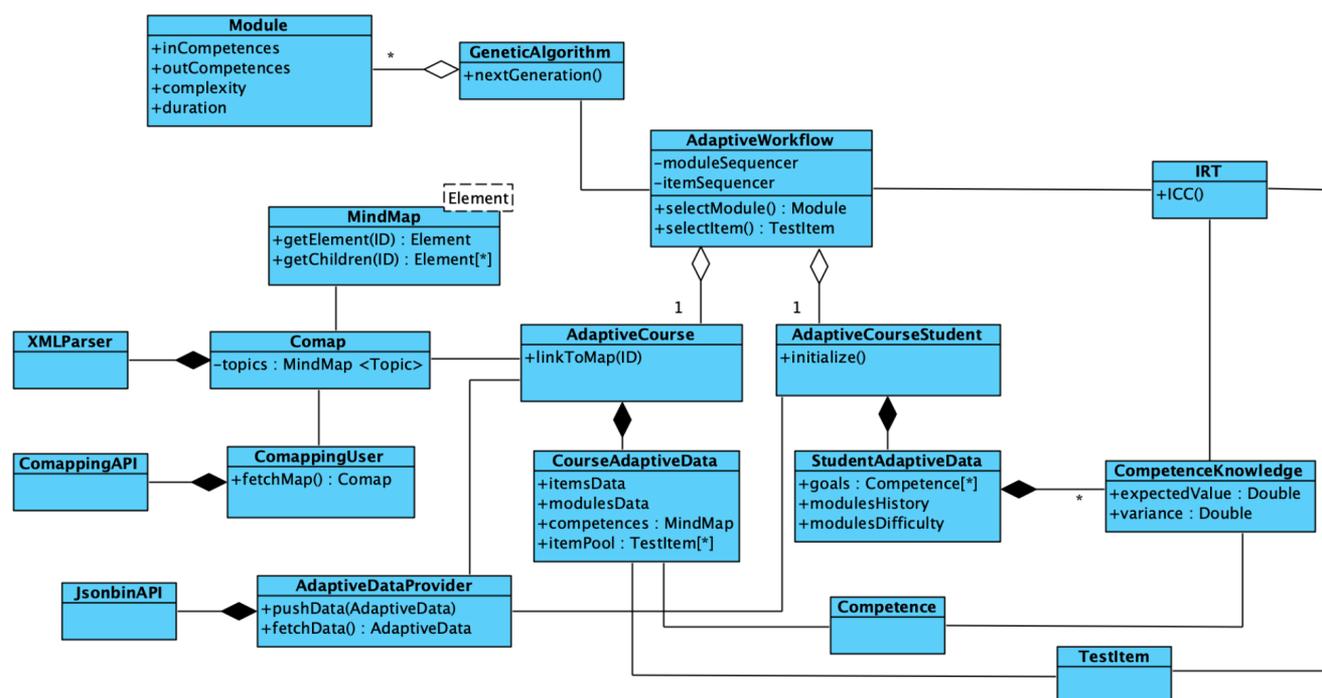
Теперь перейдем к реализации. Система реализована в виде приложения на iOS с использованием языка Swift, и основывается на следующих видах сущностей, взаимодействующих между собой:

Во-первых, это модуль – некоторая единица учебного материала. Курс состоит из набора модулей, и построение индивидуальной траектории является задачей нахождения правильной последовательности модулей.

Далее, это компетенция – набор навыков и умений, участвующих в курсе. Компетенции могут выступать требованиями к модулю и быть его результатом, что очень важно для генерации траектории.

Также, есть тесты и студент. Тесты характеризуются сложностью, задаваемой кривой ICC с тремя параметрами, а студент содержит информацию о текущем уровне освоения компетенций и некоторых других характеристиках.

# Реализация: Архитектура



8 / 14

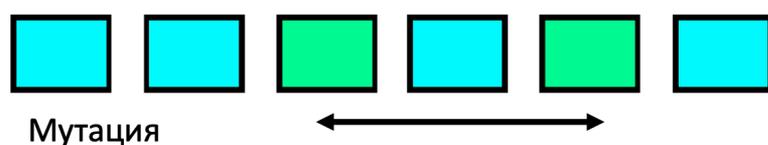
Так выглядит диаграмма классов, отражающая архитектуру приложения. За сам процесс адаптивного обучения отвечает класс AdaptiveWorkflow, делегирующий вопросы построения индивидуальной траектории и адаптивного тестирования классам ModuleSequencer и ItemSequencer соответственно. Причем ModuleSequencer использует GeneticAlgorithm в качестве алгоритмической основы, а ItemSequencer активно применяет классы IRT и CompetenceKnowledge, позволяющие оперировать оценкой знаний студента.

Классы AdaptiveCourse и AdaptiveCourseStudent занимаются адаптивными данными, связанными с курсом и со студентом, проходящим курс. С помощью класса AdaptiveDataProvider они читают и записывают измененные данные на сервер, а также сами их кэшируют.

Модули, компетенции и тесты курса хранятся в виде дерева при помощи специальной структуры MindMap.

# Реализация: Генетический алгоритм

- Генотип – последовательность модулей
- Order crossover
- Стратегия элитизма



Для реализации поиска индивидуальной траектории был избран генетический алгоритм. Последовательность модулей, которая является особью, представляется в нем числовой перестановкой. В качестве мутации используется замена друг на друга двух произвольных генов, а для скрещивания используется метод order crossover, изображенный на рисунке. Метод сохраняет блок одного из родителей и берет оставшиеся элементы в порядке, согласно второму родителю. Также реализован элитизм, то есть сохранение некоторой части популяции при переходе на следующее поколение для гарантии более качественных результатов.

# Генетический алгоритм: Функция оценки

- Моделирование прохождения траектории
  - Соблюдение входных/выходных ограничений
  - Отслеживание «пропущенных» и «просмотренных» компетенций
  - Отсевание ненужных модулей
  - Минимизация накопленной ошибки
- Минимизация общих затраченных усилий
  - Усилия = время \* сложность

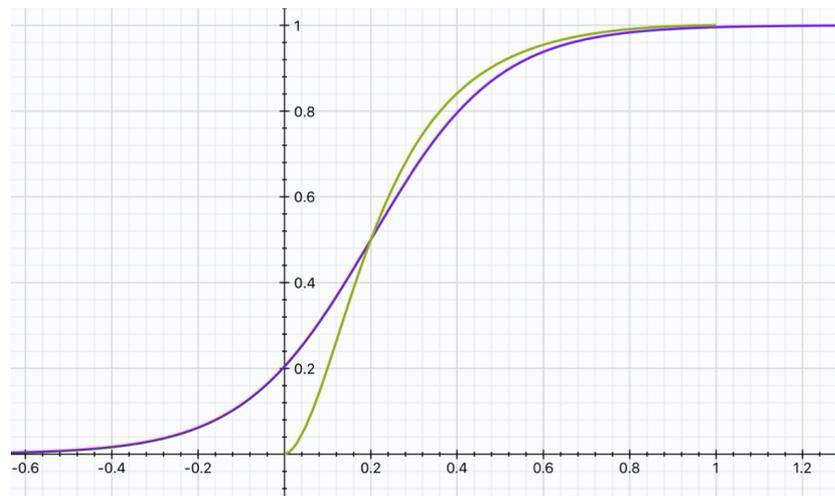
Функция оценки или функция приспособленности в генетическом алгоритме является, пожалуй, наиболее важной его частью, определяющей эффективность и работоспособность всего алгоритма. В реализованном приложении разработан подход, основанный на моделировании прохождения студентом выбранной траектории с отслеживанием на каждом шаге так называемых «пропущенных» и «просмотренных» компетенций. "Пропущенными" я называю те из них, что были неудовлетворенным требованием к какому-то из уже пройденных модулей, а «просмотренными» – те, что были не совсем честным путем получены в результате прохождения модуля, для которого не были выполнены все требования.

Помимо "просмотренных" и "пропущенных" компетенций функция также учитывает (с другим коэффициентом) общие усилия на прохождение траектории, складывающиеся из времени, рассчитанного на каждый модуль, и сложности этого модуля (где сложность модуля может корректироваться для конкретного студента). Функция останавливает вычисление в момент достижения всех возможных целевых компетенций.

Описанный генетический алгоритм был протестирован на разных начальных условиях и показывает результаты, соответствующие логическим ожиданиям, выполняясь за допустимое время.

# Реализация: Адаптивное тестирование

- Item characteristic curve
  - Логистическая функция
  - Смена области определения
- Уровень компетенции
  - Вероятностное распределение
  - Теорема Байеса



$$p_i(\theta) = c_i + \frac{1 - c_i}{1 + e^{-a_i(\theta - b_i)}}$$

11 / 14

Для адаптивного тестирования в приложении используются вероятностные алгоритмы Item Response Theory, и в качестве кривой ICC используется логистическая модель с тремя параметрами, описанная указанной здесь формулой и представленная на рисунке синим цветом. За исключением того, что кривая предварительно приводится к ограниченной области определения от 0 до 1, поскольку так показалось удобным сделать в ходе реализации. "b" означает здесь сложность задания, "a" регулирует наклон кривой ICC, а "c" – это фактор угадывания.

Уровень знаний представляется в виде дискретного вероятностного распределения, и для его корректировки используется теорема Байеса об условной вероятности. То есть, после каждого верного или неверного ответа вероятности пересчитываются с учетом нового условия о правильном или неправильном ответе на вопрос.

# Реализация: Организация данных

- Comapping – онлайн-система создания и редактирования MindMap
- Comapping API: Хранение карт в XML-формате
  - Курс / компетенции / тесты
  - SWXMLHash
- Jsonbin API: Хранение мета-данных в JSON-формате
  - Модель студента
  - Разметка курса

```
<node id="1058529" priority="1">
  <note><![CDATA[This is a note]]></note>
  <text><![CDATA[Use this comap to learn the basics]]></text>
  <node arrow="blue" flag="caution" id="1058530" smiley="happy"
star="yellow" taskCompletion="50">
    <icon name="clock"/>
    <task deadline="tomorrow" estimate="1 day" responsible="You"
start="today"/>
    <text><![CDATA[Don't worry - you can't break anything no
matter how hard you try!]]></text>
  </node>
</node>
```

XML-представление карты

12 / 14

Для хранения материалов курса, компетенций и тестов используется система Comapping – онлайн система создания и редактирования интеллектуальных карт. Она позволяет удобно структурировать информацию, а также прикреплять к узлам произвольные файлы, и поэтому неплохо подходит для создания онлайн-курсов.

Comapping используется для хранения двух карт, одна из которых представляет материалы курса, а вторая – его компетенции и тесты для компетенций.

Comapping хранит карты в формате XML и предоставляет API, позволяющее использовать себя внутри приложения. Для серверного хранения всей остальной мета-информации о курсе и студентах, которые его проходят, используется другой сервис Jsonbin, предлагающий простой REST-API и использующий JSON-формат.

# Эксперименты



Карта компетенций курса

Какой буфер для ребер используется в методе M2?

DF

UR

UB

DB

Quit Guess Next

Пример теста

В качестве эксперимента был создан адаптивный курс по спидкубингу, то есть сборке кубика Рубика на скорость. Курс был предложен для прохождения двум людям, один из которых был довольно поверхностно ознакомлен с этой темой, а второй занимается достаточно профессионально. Ими изначально был выбран одинаковый набор целевых компетенций. В результате прохождения курса и корректировки индивидуальных показателей в процессе прохождения, для одного из них траектория обучения получилась значительно короче, чем для другого, что говорит об адаптивности системы.

# ИТОГИ

Разработана система адаптивного обучения, позволяющая создавать и проходить адаптивные курсы:

- Исследованы существующие разработки и алгоритмы в области адаптивного обучения
- Разработана архитектура приложения
- Подобраны и реализованы алгоритмы поиска образовательной траектории и адаптивного тестирования
- Реализована система адаптивного обучения с использованием подобранных алгоритмов
- Построен адаптивный курс и проанализированы результаты его прохождения реальными людьми

Таким образом, в результате была разработана система адаптивного обучения, позволяющая создавать и проходить адаптивные курсы, и были выполнены все поставленные задачи.

На этом все, спасибо за внимание.