



Описание прототипа системы

Прототип системы организован в виде клиент-серверного приложения. Для разработки прототипа используется язык Java.

Клиентская часть представляет собой мобильное приложение под управлением операционной системы Android и реализует следующий функционал:

- задание анализируемого сигнала;
- выбор базисного вейвлета из предоставленного набора;
- задание параметров преобразования;
- отправка данных в формате JSON на сервер и получение результатов от него;
- визуализация результатов.

Поскольку расчет вейлет-преобразования является достаточно ресурсоемкой задачей, а мобильные устройства обладают более скромной аппаратной базой, чем стационарные ЭВМ, решение этой задачи возложено на серверную часть прототипа приложения.

Заключение

В результате работы создан прототип системы, реализующей вейвлет-анализ одномерных сигналов. Прототип обеспечивает базовую функциональность для проведения вейвлет-анализа и позволяет задать анализируемый сигнал, параметры преобразования, базисный вейвлет, выполнить расчет преобразования и визуализацию результатов.

В перспективе необходимо предусмотреть ряд возможностей, таких как: авторизация клиента, сохранение результатов и соответствующих им параметров в базу данных, возможность анализа двумерных сигналов, трехмерная визуализация результатов.

Литература

1. Астафьева, Н.М. Вейвлет-анализ: основы теории и примеры применения [Текст] / Н.М. Астафьева // Успехи физических наук. -1996. -Т. 166. -Вып. 11. -С. 1145-1170.
2. Нагорнов, О.В. Вейвлет-анализ в примерах [Текст]: учебное пособие / О.В. Нагорнов, В.Г. Никитаев, В.М. Простокишин, С.А. Тюфлин [и др.]. -М.: НИЯУ МИФИ, 2010. -120 с.



С.А. Прохоров, С.А. Сучкова, Е.В. Матыцин, И.М. Куликовских

СИСТЕМА АДАПТИВНОГО ОБУЧЕНИЯ НА ОСНОВЕ КОЛЛАБОРАТИВНОЙ ФИЛЬТРАЦИИ

(Самарский национальный исследовательский университет
имени академика С.П. Королёва)

Применение технологий анализа данных в области образования привело к формированию новой образовательной модели – адаптивного обучения. Основная идея, положенная в основу адаптивного обучения [1-6], заключается в персонализации процесса обучения через формирование индивидуальных обучающих траекторий. Как правило, в качестве исходных данных для построения обучающей траектории используется следующая информация [7,8]:

- цели и алгоритмы взаимодействия группы обучаемых;
- алгоритмы обработки данных в реальном времени для формирования системы рекомендаций;
- формирование признаков, отражающих знания и умения обучаемых;
- формирование признаков, отвечающих за особенности восприятия информации;
- формирование признаков, оценивающих чувствительность обучаемых к изменениям в преподавании, темпе, обучающем контенте и т.д.

Основным результатом построения обучающей траектории является адаптация процесса обучения на основе следующих количественных показателей:

- статистика успеваемости каждого обучаемого, учитывающая историю обучения;
- результаты предсказания успеваемости, скорости и вероятности достижения цели, а также ожидаемую оценку и уровень знаний.

Следует отметить, что помимо количественных показателей, индивидуальная траектория позволяет проанализировать обучающий контент по множеству показателей: специфика выделения ключевых идей, анализ структуры материала, уровень его сложности, формат и т.д.

Предлагаемые ранее системы адаптивного обучения, главным образом, направлены на анализ демонстрируемых знаний обучаемых с целью их дальнейшего развития. Однако, знания могут быть частичными и, таким образом, выраженными неявно. Одним из способов контроля качества знаний, при котором приходится сталкиваться с проблемой оценивания частичных знаний, является тестирование с множественным выбором. Помимо двух простейших исходов, когда обучаемый знает правильный ответ и отвечает правильно либо не знает и пропускает, могут быть следующие варианты:



- знает правильный ответ, но отвечает неправильно (например, в результате влияния внешних факторов, таких как фактор случайности, некорректно составленное задание и т.д.)

- не знает правильного ответа, но может исключить некоторые некорректные варианты;

- не знает правильного ответа и исключает из рассмотрения правильный ответ как некорректный;

- не знает правильного ответа и пытается угадать.

Более того, на представленные варианты могут налагаться дополнительные ограничения, связанные с условиями проведения тестирования:

- наличие «пенальти» за неправильный ответ;

- возможность обсуждать задания и вероятность правильности того или иного варианта.

При таких условиях, задача оценивания частичных знаний становится еще более неоднозначной и складывается из ряда психологических факторов, включающих предрасположенность обучаемых гадать, рисковать или полагаться на мнение окружающих.

Таким образом, целью данной работы является создание системы адаптивного обучения, которая анализирует частичные знания обучаемых с учетом описанных выше особенностей при тестировании с множественным выбором.

Формализация задачи

В соответствии с поставленной целью необходимо формализовать задачу создания системы адаптивного обучения и предложить математический аппарат для ее решения.

Представим стратегию поведения обучаемого при тестировании с множественным выбором в виде следующей структуры, которая отражает логику дерева принятия решений:

--читает тестовое задание:

----пропускает, переходит к следующему заданию:

-----**не знает ответ.**

-----знает ответ, но не уверен.

-----отвечает:

-----**знает ответ.**

-----не знает, но пытается угадать;

-----не знает, но спрашивает совет:

-----полагается на мнение других.

-----полагается на собственное мнение.

В представленной структуре жирным выделены простейшие бинарные исходы ($y=0$ и $y=1$). Остальные признаки должны быть закодированы в виде пары (x, y) , где вектор x представляет собой набор признаков, описывающих стратегию поведения обучаемого при тестировании. Тогда, задача построения обучающей траектории сводится к нахождению $p(y=1|x;\theta)$ для оптимизации



стратегии обучения, что представляет собой задачу оптимизации коэффициентов θ логистической регрессии $h_\theta(x)$.

В терминах машинного обучения данная задача может интерпретироваться как задача *online machine learning* [9,10]. В качестве базового алгоритма оптимизации для данного класса задач используется градиентный спуск. Тогда, представим задачу нахождения оптимальной обучающей траектории в виде следующего псевдокода:

```
Repeat {
  Get  $(x, y)$  corresponding to a learner
  Update  $\theta$  using  $(x, y)$ :
     $\theta_j = \theta_j - \alpha(h_\theta(x) - y)x_j$ ,  $(j=0\dots n)$ ,  $\alpha$  - learningrate
}
```

Следует отметить, что в отличие от классической постановки задачи оптимизации коэффициентов регрессии, где анализируется выборка значений (x_i, y_i) , в данной задаче используется вектор (x, y) , соответствующий одному обучаемому, который постоянно дополняется новыми признаками в течение всего процесса обучения.

Для учета результатов влияния коллаборативного принятия решения, дополним данную задачу в терминах другого класса задач машинного обучения – *collaborative filtering* [11-13].

Пусть n_i, n_j – количества обучаемых и тестов соответственно. Тогда, $r(i, j)=1$, если j -ый обучаемый отвечал на i -ое задание теста, а $y(i, j)$ – количество правильных ответов, которые j -ый обучаемый дал на i -ое задание теста. Задача оптимизации обучающей траектории в терминах *collaborative filtering* может быть представлена в виде следующего псевдокода:

```
Initialize  $x^{(1)}, \dots, x^{(n)}$ ,  $\theta^{(1)}, \dots, \theta^{(n)}$  to small random values
Repeat for each  $j=1, \dots, n_j$ ,  $i=1, \dots, n_i$ 
{
  Simultaneously update  $x^{(i)}$ ,  $\theta^{(j)}$ :
     $x_k^{(i)} = x_k^{(i)} - \alpha \sum_{j:r(i,j)=1} [h(\theta^{(j)})^T x^{(i)} - y^{(i,j)}] \theta_k^{(j)}$ ,  $\alpha$  - learning rate
     $\theta_k^{(j)} = \theta_k^{(j)} - \alpha \sum_{i:r(i,j)=1} [h(\theta^{(j)})^T x^{(i)} - y^{(i,j)}] x_k^{(i)}$ 
}
```

Коллаборативная фильтрация позволяет решать задачу одновременной оптимизации обучающей траектории для данного обучаемого с признаками $\theta^{(j)}$ и предоставляемого контента $x^{(i)}$. Более того, на основе гипотезы $h(\theta^{(j)})^T x^{(i)}$



может выполняться прогноз успеваемости, скорости и вероятности достижения цели.

Данную постановку можно также использовать для нахождения меры сходства между ответами группы обучаемых на тестовые задания [13] и таким образом выявлять полагался ли обучаемый на мнение других при ответе на данное тестовое задание.

Программная реализация

Для реализации предложенных математических моделей, а также с целью сбора необходимых признаков, характеризующих поведения обучаемых при тестировании, была спроектирована система адаптивного обучения.

В качестве базовой архитектуры при создании системы использовалась архитектура стандартного *Java EE application*, имеющая следующие уровни: *Web*, *Business*, *Data*. *Web tier* – уровень представления, отвечающий за показ данных пользователю. *Business tier* – уровень бизнес логики, отвечающий за обработку информации согласно некоторым бизнес правилам. *Data tier* – уровень данных, отвечает за хранение информации. Данные уровни абстракции согласуются с постановкой задачи и позволяют создать сложную систему с необходимым уровнем масштабируемости и расширяемости.

Рассмотрим физическую реализацию, где для получения полной абстракциуровень представления выделен на отдельный сервер (рис. 1).

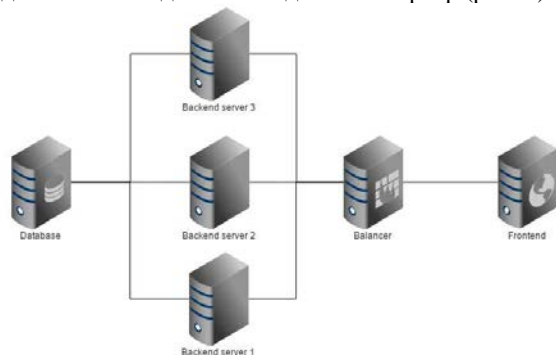


Рис. 1. Физическая реализация

Работа данной схемы будет происходить следующим образом. Пользователь имеет доступ только к *“Frontend”* серверу. Визуальный интерфейс по специальному API будет работать только с одним сервером *“Balancer”*, который не имеет никакой бизнес логики и предназначен только, для распределения нагрузки между *“Backend”* серверами. Полученные запросы обрабатываются на *“Backend”* сервере. При необходимости выполняются дополнительные запросы в базу данных, после чего ответ отправляется обратно *“Frontend”* серверу. Подобная архитектура позволяет распределить нагрузку на всю систему в целом, что, в свою очередь, увеличивает скорость работы в целом.



Рассмотрим вспомогательные технологии, которые мы будем использовать на каждом уровне (см. рис. 2).

В качестве базы данных была выбрана MySQL как легкая, реляционная база данных, отвечающая требованиям постановки задачи. В свою очередь, библиотека Hibernate позволяет поддерживать персистентность между таблицами, реализуемых в виде сущностей. Фреймворк Spring предоставляет API по работе контроллерами, фильтрами, валидацией и другими интерфейсами, которые подготавливают данные к отправке на клиент или собирают информацию с клиента.

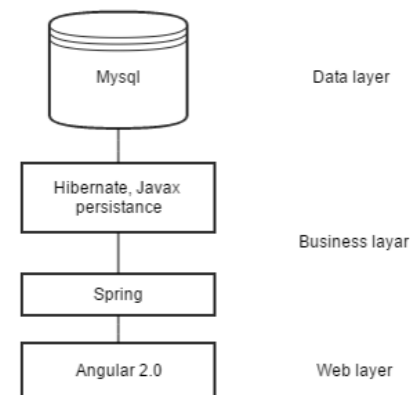


Рис. 2. Технологии, использованные на каждом уровне

Схема таблиц и их связей представлена на следующей диаграмме (см. рис. 3), где

- User Entity – сущность для связи с таблицей “users”, для хранения информации о пользователе;
- Test Entity - сущность для связи с таблицей “tests”, для хранения информации о тесте;
- Test Relation Entity – вспомогательная сущность для связи таблиц “users” и “tests”;
- Test Question Entity - сущность для связи с таблицей “questions”, для хранения информации о вопросе из теста;
- Test Multi Choice Entity - сущность для связи с таблицей “choices”, для хранения информации о возможных выборах ответа в вопросе;
- Test Record Entity – сущность для связи с таблицей “test-records”, для хранения информации собранной при прохождении теста пользователем;



- Question Record Entity – сущность для связи с таблицей “questions-records”, для хранения информации собранной при ответе пользователем на отдельный вопрос;
- Multi Choice Record Entity – сущность для связи с таблицей “choice-records”, для хранения информации о ответе на вопрос пользователем.

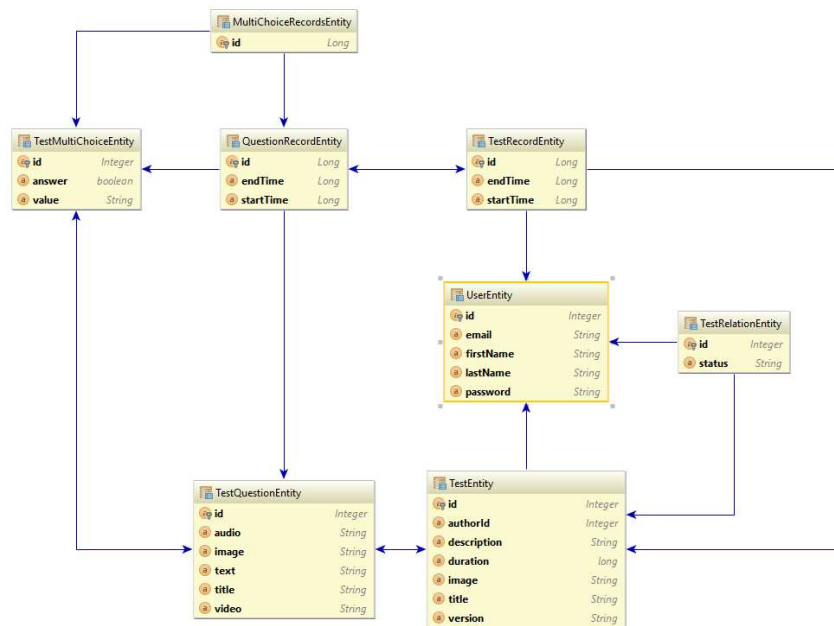


Рис. 3. Схема таблиц и их связей

Литература

1. P. Baepler, C.J. Murdoch *Academic Analytics and Data Mining in Higher Education*, International Journal for the Scholarship of Teaching and Learning, 2(4), 2010,
2. B. Dietz-Uhler & J.E. Hurn *Using Learning Analytics to Predict (and Improve) Student Success: A Faculty Perspective*, Journal of Interactive Online Learning, 1(12), 2013, URL: www.ncolr.org/jiol
3. Brijesh Kumar Baradwa, Saurabh Pal *Mining Educational Data to Analyze Students' Performance*, International Journal of Advanced Computer Science and Applications, 6(2), 2011, pp. 63-69.
4. *Adaptive Learning: An insider's view of how adaptive learning works*(2015),URL: <http://www.knewton.com/blog/adaptive-learning/> (дата обращения 27 марта 2016).



5. A. Peña-Ayala *Educational data mining: A survey and a data mining-based analysis of recent works*, Research in Higher Education Journal, Expert Systems with Applications, 4(41), 2014, pp. 1432-1462.
6. А.В. Топчиев, В.А. Чулюков *Модели адаптивного обучения в компьютерных системах*, Современные наукоемкие технологии, Технические науки, №5, 2010, С. 62-68.
7. *KNEWTON: Образование на уровне атомов*(2013),URL: <http://www.edutainme.ru/post/knewton/>(дата обращения 27 марта 2016).
8. *Что такое адаптивное обучение*(2014),URL: <http://www.edutainme.ru/post/что-такое-адаптивное-обучение/>(дата обращения 27 марта 2016).
9. Shai Shalev-Shwartz *Online Learning and Online Convex Optimization*, Foundations and Trends in Machine Learning, Vol. 4, No. 2, 2011, pp. 107-194.
10. J. Duchi, E. Hazan, Y. Singer *Adaptive Subgradient Methods for Online Learning and Stochastic Optimization*,
11. M.D. Ekstrand, J.T. Riedl, J.A. Konstan *Collaborative Filtering Recommender Systems*, Foundations and Trends in Human-Computer Interaction, Vol. 4, No. 2, 2010, pp. 81-173.
12. P. Melville, R. J. Mooney, R.Nagarajan *Content-Boosted Collaborative Filtering for Improved Recommendations*, Proceedings of the Eighteenth National Conference on Artificial Intelligence (AAAI-2002), pp. 187-192, July 2002, Edmonton, Canada.
13. M. Weimer, A. Karatzoglou, A. Smola *Adaptive Collaborative Filtering*, RecSys'08, pp. 275-282, October 23-25, 2008, Lausanne, Switzerland.

Н.К. Федюнин, И.В. Лёзина

ПРОГНОЗИРОВАНИЕ ВРЕМЕННЫХ РЯДОВ РАДИАЛЬНО-БАЗИСНОЙ СЕТЬЮ

(Самарский национальный исследовательский университет
имени академика С.П. Королёва)

Временной ряд (или ряд динамики) — собранный в разные моменты времени статистический материал о значении каких-либо параметров (в простейшем случае одного) исследуемого процесса. Временной ряд существенно отличается от простой выборки данных, так как при анализе учитывается взаимосвязь измерений со временем, а не только статистическое разнообразие и статистические характеристики выборки [1]. Прогнозирование рядов динамики является распространенной задачей, решение которой несомненно важно не только экономистам, но и обычным людям.

Способности нейронных сетей к прогнозированию напрямую следуют из их способности к обобщению и выделению скрытых зависимостей между входными и выходными данными. После обучения сеть способна предсказать бу-