

Определение типов ошибок в решениях задач онлайн-курсов по программированию

Лобанов Артём Владиславович

Научный руководитель: к.т.н., доц. Брыксин Тимофей Александрович

Рецензент: Сартасов Станислав Юрьевич

Национальный исследовательский университет «Высшая школа экономики»
Санкт-Петербург



Достоинства и недостатки онлайн-курсов

Плюсы:

- Массовость
- Доступность
- Накопление базы решений

Минусы:

- Слабая обратная связь



Цели и задачи

Найти способ автоматически выдавать пользователям подсказки к исправлению решений

1. Изучить существующие решения
2. Предложить подход к автоматической выдаче подсказок
 - a. Научиться выделять частые ошибки
 - b. Научиться находить эти ошибки в новых решениях
3. Подобрать оптимальные параметры решения
4. Исследовать качество решения



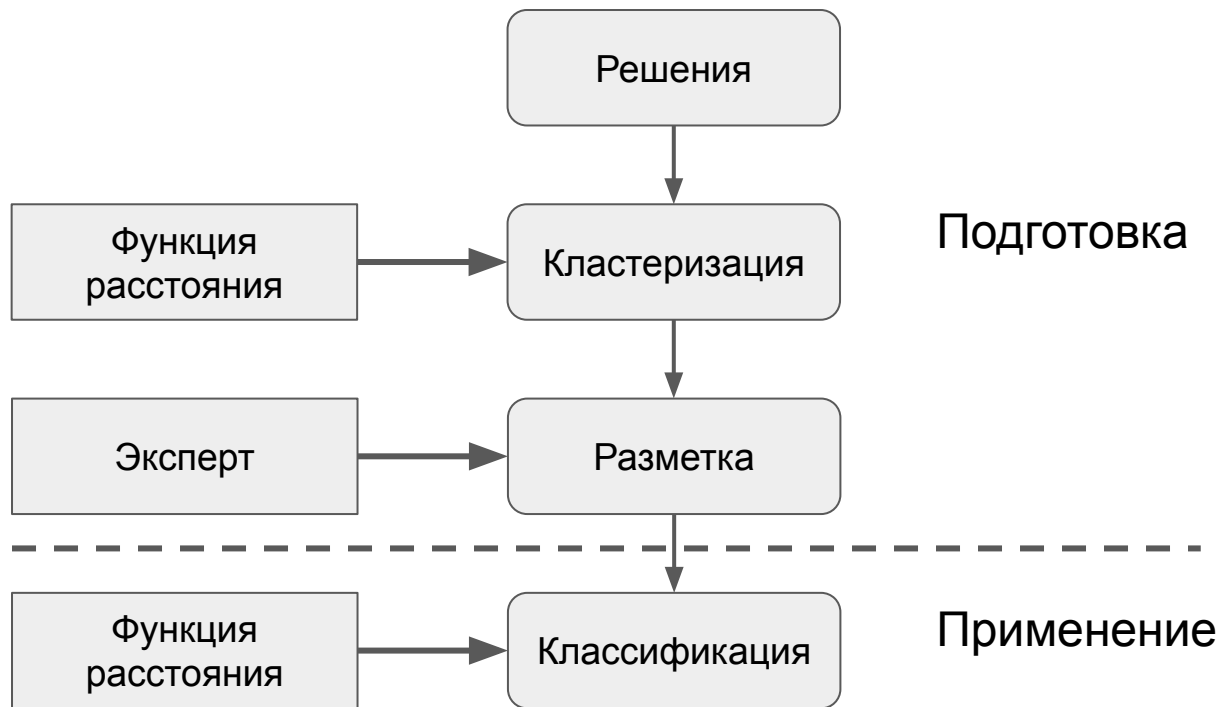
Существующие решения

- Определение “запахов кода”
- Прогнозирование ошибок
- Анализ путей в пространстве решений^[1,2]

¹ Barnes et al., Toward automatic hint generation for logic proof tutoring using historical student data, Springer, 2008

² Rivers et al., Data-driven hint generation in vast solution spaces: a self-improving python programming tutor, Springer, 2017

Подход к решению





Выделение изменений

- Анализ изменений программного кода
 - Построчная разница
 - Преобразования AST (лучшая корреляция с наличием ошибок¹)
- Для каждого неправильного решения ищется ближайшее правильное
- Для выделения изменений использовалась библиотека GumTree

¹ Giger Emanuel et al., Comparing fine-grained source code changes and code churn for bug prediction, ACM, 2011



Решение

- Кластеризация
 - Иерархическая агломеративная кластеризация^[1, 2] (НАС)
- Классификация
 - Поиск ближайшего кластера (НС)
 - Метод k ближайших соседей^[3] (k-NN)

¹Kreutzer et al., Automatic clustering of code changes, ACM, 2016

²Fluri et al., Discovering patterns of change types, IEEE, 2008

³Osman et al., Hyperparameter optimization to improve bug prediction accuracy, IEEE, 2017



Функция расстояния

- Расстояние Жаккара
- Подход “Мешок слов” для изменений
- Векторизация изменений
- Нечёткая модификация расстояния Жаккара



Данные

- 6 реальных задачи со Stepik (язык решений - Java)
- Ручная разметка данных (800 примеров для каждой задачи)
- Два решения из каждой сессии: правильное и неправильное

Задача	Обучающая	Обучающая(размечено)	Тестовая (размечено)
A	8683	600	200
B	3203	600	200
C	4017	600	200
D	2424	600	200
E	2076	600	200
F	12707	600	200



Подбор параметров

- 4 задачи
- PR-AUC в качестве метрики
- k-блочная кросс-валидация

d_{hac}	t_{hac}	M	d_c	Классификатор	A	B	C	D	Среднее
bow	0.3	40	fuz_jac	k-NN (k=15)	0.770	0.858	0.793	0.713	0.784
fuz_jac	0.5	40	fuz_jac	k-NN (k=10)	0.773	0.839	0.793	0.675	0.770
def_vec	0.4	40	fuz_jac	k-NN (k=15)	0.730	0.682	0.815	0.672	0.725
bow	0.3	40	fuz_jac	k-NN (k=10)	0.773	0.852	0.786	0.718	0.782

fuz_jac - нечёткое расстояние Жаккара, эвристическая формула схожести элементарных изменений

bow - подход “мешок слов”

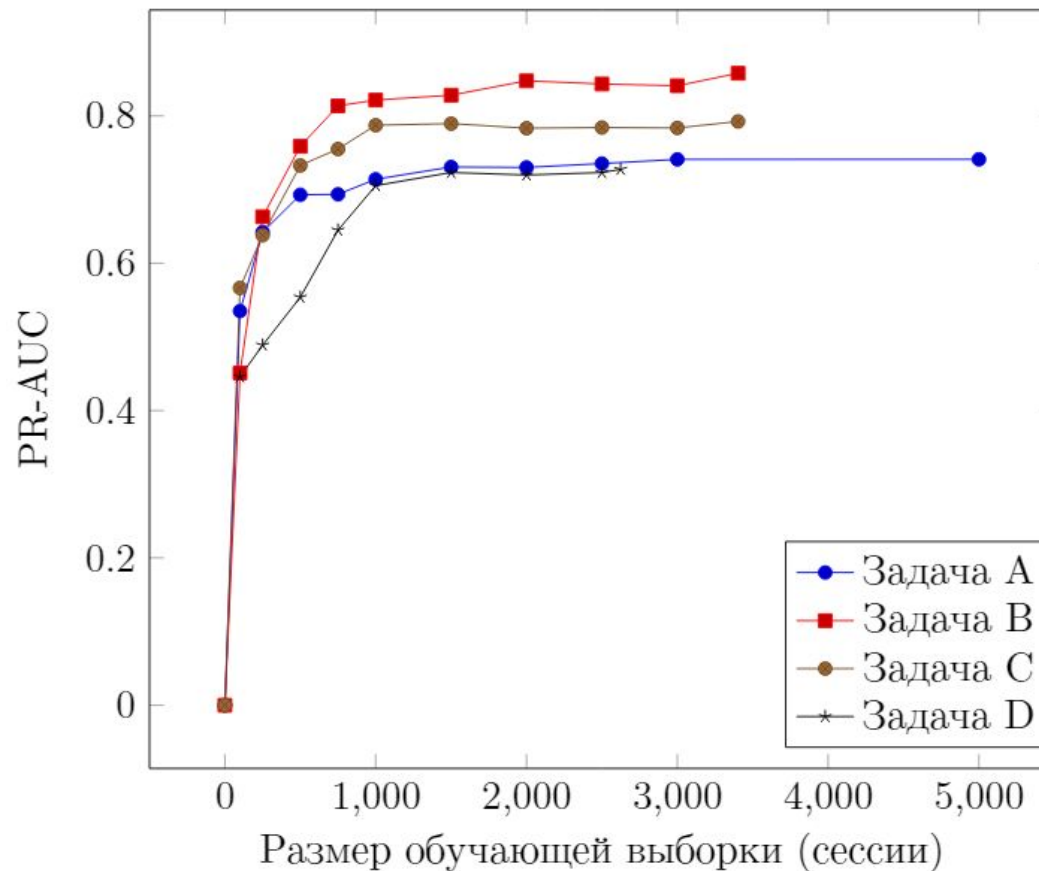
def_vec - нечёткое расстояние Жаккара, определение схожести элементарных изменений с помощью их векторизации



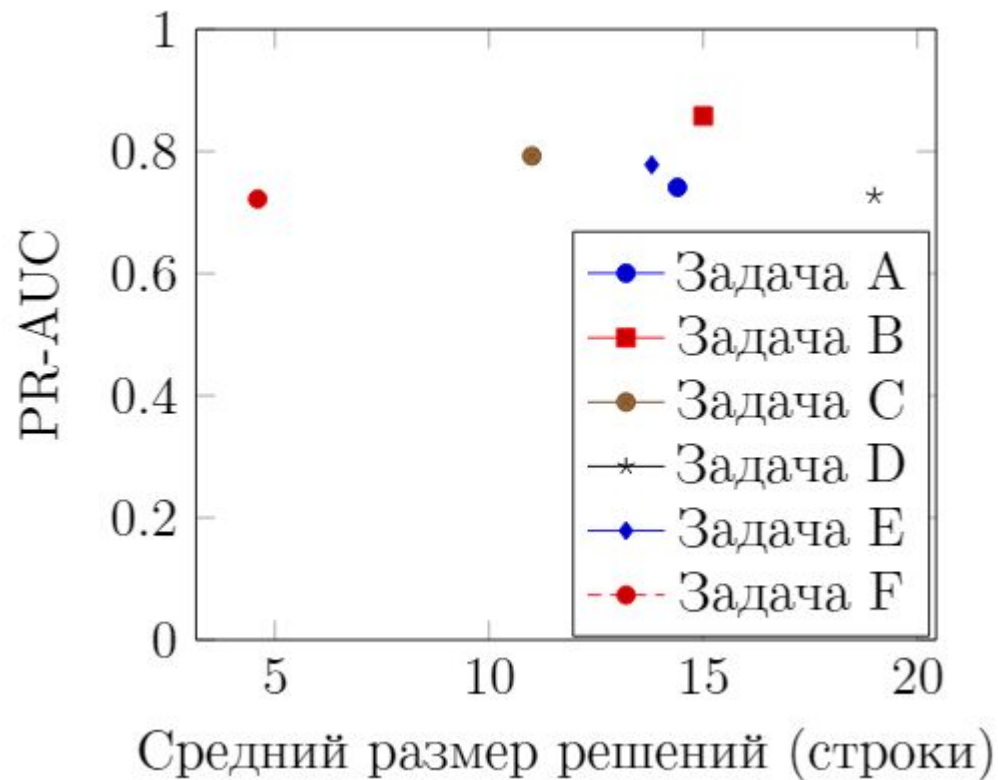
Независимое тестирование

Задача	Кросс-валидация	Тестовая выборка
A	0.770	0.741
B	0.858	0.857
C	0.793	0.793
D	0.713	0.727
E	-	0.778
F	-	0.722

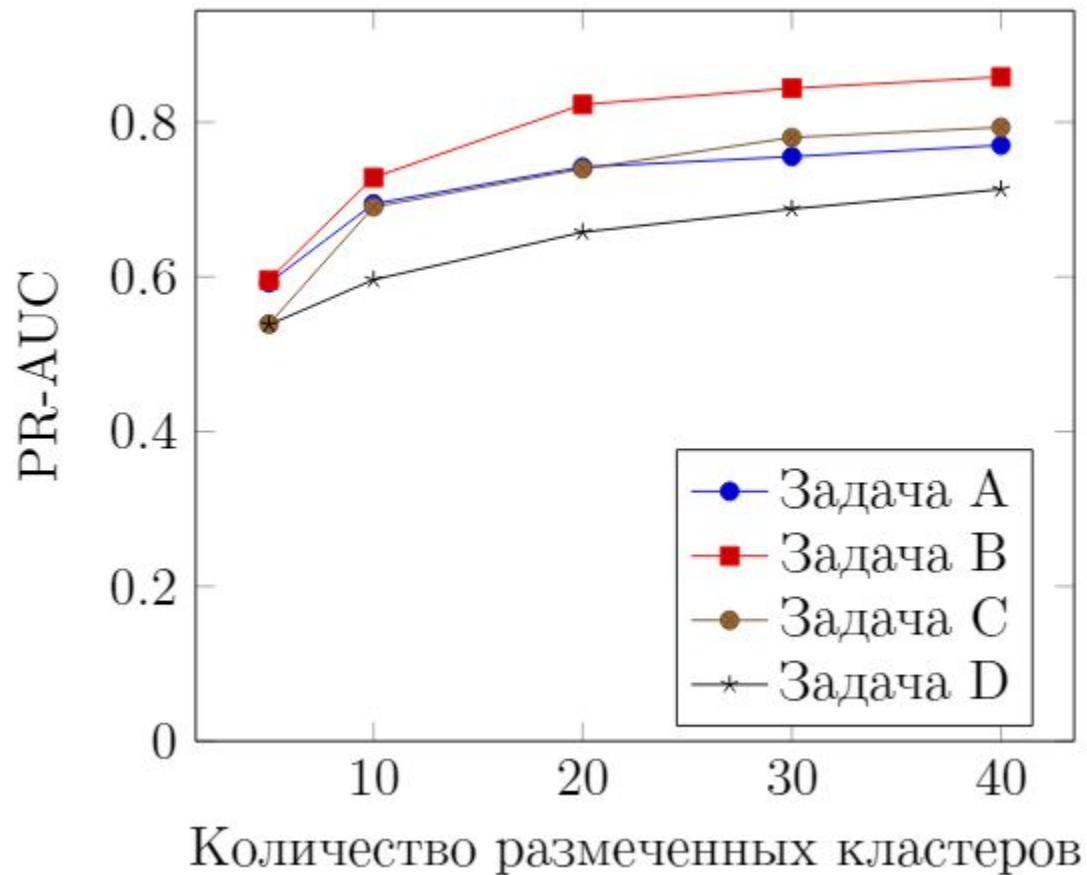
Зависимость от размера выборки



Зависимость от размера решений



Зависимость от качества разметки





Результаты

- Предложен подход к решению проблемы, основанный на анализе сценариев редактирования
- Проведено исследование качества решения
- Исследованы границы применимости
- Результаты исследования будут представлены на The 20th International Conference on Artificial Intelligence in Education (AIED'19)

Задача	Тестовая выборка (PR-AUC)
A	0.741
B	0.857
C	0.793
D	0.727
E	0.778
F	0.722