Синтез программного кода с использованием машинного обучения

А. Е. Чебыкин, 471
Научный руководитель: Я. А. Кириленко, ст. преп.
Рецензент: А. А. Шпильман, СПбАУ РАН ст. преп.

Мотивация

- Некоторые шаги разработки уже автоматизируются
- Генерация программы для работы с API библиотеки по описанию на естественном языке
- Исследуются разные подзадачи
- Можно применить модели машинного обучения

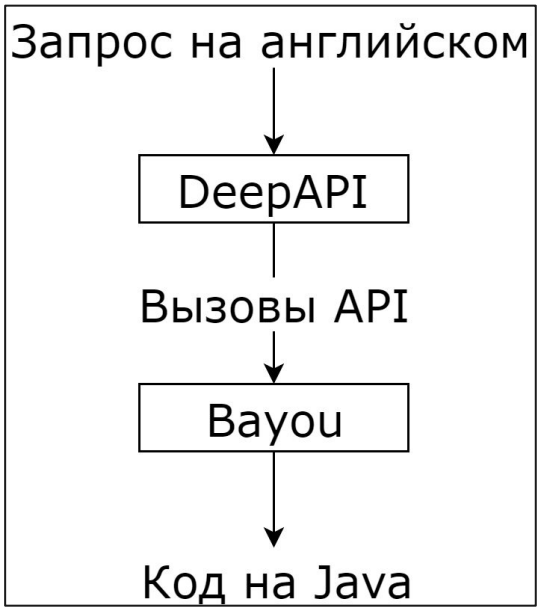
Рекомендация вызовов API

- Как работать с библиотекой?
- MAPO, UP-Miner
  - Кластеризация вызовов API
- DeepAPI
  - Предложение цепочки API по текстовому запросу
    - “write text to file” – “FileWriter.new FileWriter.write FileWriter.close”
  - Перевод с английского на язык вызовов API
  - Рекуррентные нейронные сети
  - DeepAPI# (SYRCoSE 2018)

Генерация кода

- Ying et al
  - На вход принимает описание на естественном языке
  - Непрактично длинные описания каждой строки
    - ”for i in range(0, len(result)): ” -
      ”for every i in range of integers from 0 to length of result, not included”
- Bayou
  - На вход принимает вызовы API
  - Генерирует отрывки кода

Предлагаемое решение: DeepAPI + Bayou



1. Write text to file

2. FileWriter.new FIleWriter.write FileWriter.close

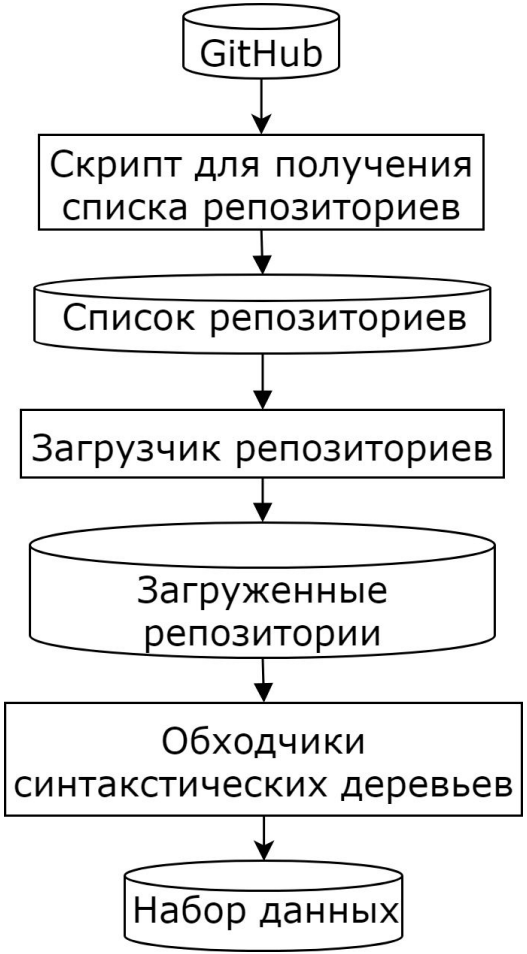
3.
```
void write(String path) {
    FileWriter fw1;
    try {
        fw1 = new FileWriter(path);
        fw1.write(path);
        fw1.close();
    } catch (IOException _e) {}
    return;
}
```

Задачи

- Собрать данные для обучения модели DeepAPI
- Эффективно обучить модель
- Разработать интеграцию с существующей IDE в виде плагина
- Апробировать плагин

Сбор данных

- GitHub
- 400,475 проектов
- Разработан набор инструментов
  - JavaParser
  - SQLite
- 17 631 306 пар данных



Извлечение данных

```java
/**
 * Copies bytes from a large (over 2GB) <code>InputStream</code> to an
 * <code>OutputStream</code>.
 * ...
 * @since 2.2
 */
public static long copyLarge(final InputStream input,
        final OutputStream output, final byte[] buffer) throws IOException {
    long count = 0;
    int n;
    while (EOF != (n = input.read(buffer))) {
        output.write(buffer,  off: 0, n);
        count += n;
    }
    return count;
}
```

Последовательность API: InputStream.read OutputStream.write
Описание: copies bytes from a large inputstream to an outputstream

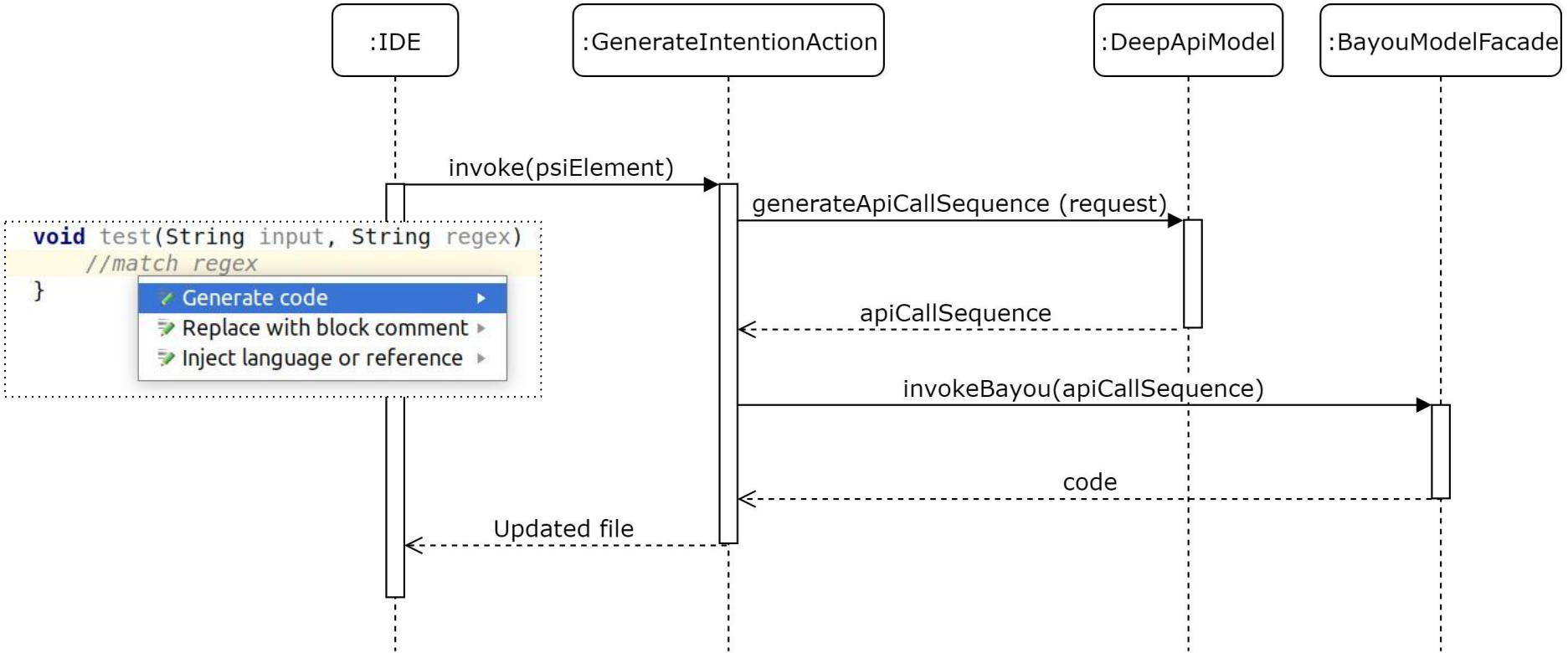
Обучение модели

- Предобработка данных
  - Количество звёзд репозитория
  - Определение языка комментария
  - Фильтрация по словарю
  - Сокращение повторяющихся вызовов
  - Удаление дубликатов
  - Фильтрация по популярности
  - Итоговое количество данных: 512 183
- OpenNMT-tf
  - Библиотека для обучения моделей машинного перевода
  - Существует Java API

Производительность DeepAPI

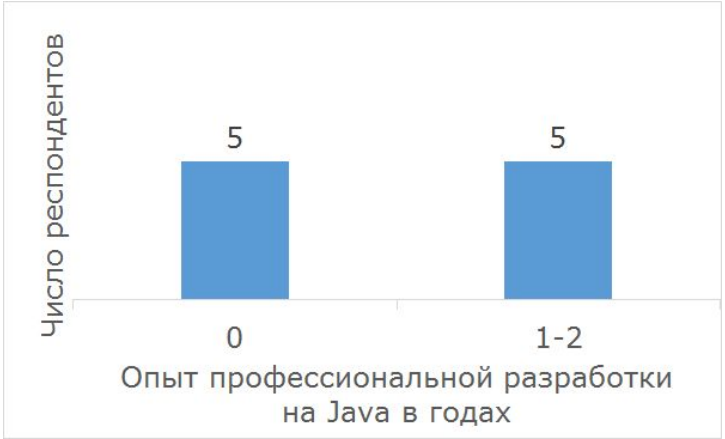
- Метрика BLEU
  - Процент совпадающих n-грамм
  - С исходными параметрами - 3.46
  - С предобработкой - 8.12
  - С улучшенными параметрами - 12.82 (47.73)
- Тестирование на 25 популярных запросах

Плагин

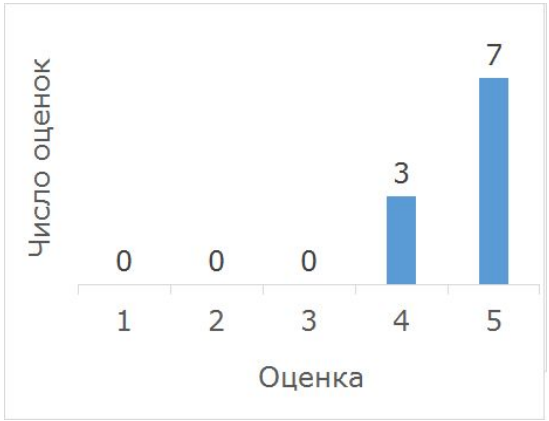

:IDE
:GenerateIntentionAction
:DeepApiModel
:BayouModelFacade
invoke(psiElement)
generateApiCallSequence (request)
apiCallSequence
invokeBayou(apiCallSequence)
code
Updated file
void test(String input, String regex)
//match regex
}
Generate code
Replace with block comment
Inject language or reference

Апробация плагина

- 10 начинающих разработчиков
- 5 заданий
  - Регулярные выражения
  - Чтение файла
  - Файлы в папке
  - Строка из числа
  - Число из строки

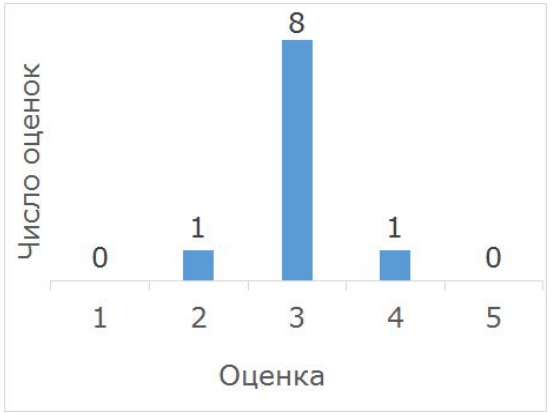


Результаты апробации



UI

Полезность



Скорость

Результаты апробации

Результаты

- Сделан обзор области генерации программного кода
- Реализованы инструменты для сбора данных
- Обучена модель DeepAPI с оптимальными параметрами и предобработкой данных
- Разработан плагин для IntelliJ IDEA
- Плагин протестирован пользователями, которые оценили его в целом положительно