



Ассистент для разработчиков в среде IntelliJ IDEA

Автор: Кита Михаил Евгеньевич, 471 группа

Научный руководитель: к.т.н., доцент Т. А. Брыксин

Рецензент: руководитель направления «Машинное обучение и анализ данных», ООО «Интеллиджей Лабс» А. А. Шпильман

Введение



- IDE предоставляют множество инструментов для облегчения труда программистов
- Иногда существующих инструментов недостаточно, приходится покидать IDE
- Решение: сделать полезную информацию доступной непосредственно в среде разработки

Постановка задачи



- Цель: расширение для IntelliJ IDEA, предоставляющее полезную информацию для решения текущих проблем
- Задачи:
 - провести анализ предметной области и существующих решений
 - реализовать алгоритм получения полезной информации
 - разработать архитектуру инструмента
 - реализовать пользовательский интерфейс
 - провести апробацию разработанного решения

Основные подходы



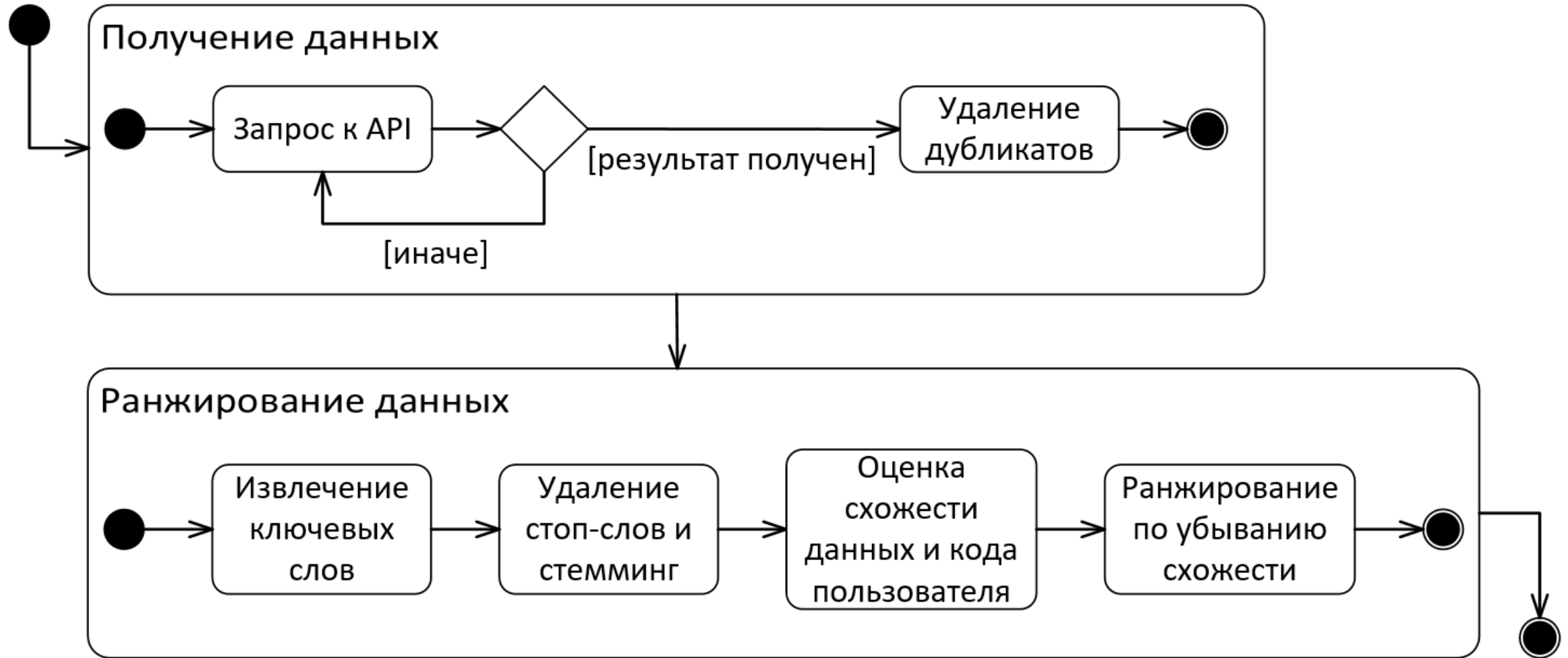
- Генеративный
 - преобразование описания задачи в полезные данные
 - пример: автоматическая генерация кода
- Ранжирующий
 - переиспользование данных, созданных другими людьми
 - пример: рекомендательные системы

Существующие решения



| Инструмент | Платформы | ЯП | Поддержка нового ЯП | Текстовые запросы | Автомат. подсказки | Помощь с ошибками |
|-----------------|-----------------------------|------------|---------------------|-------------------|--------------------|-------------------|
| Codota | Eclipse, IntelliJ IDEA | Java | Данные, обучение | Да | Нет | Нет |
| Kite | Atom, Sublime Text, PyCharm | Python | Данные, обучение | Да | Нет | Да |
| BDA | Visual Studio | C, C++, C# | Данные | Да | Нет | Да |
| NLP2Code | Eclipse | Java | | Да | Нет | Нет |
| Selene | Eclipse | Java | Данные | Нет | Да | Нет |
| Seahawk | Eclipse | Java | Данные | Да | Нет | Нет |
| Prompter | Eclipse | Java | | Да | Да | Нет |

Алгоритм рекомендации



Получение данных



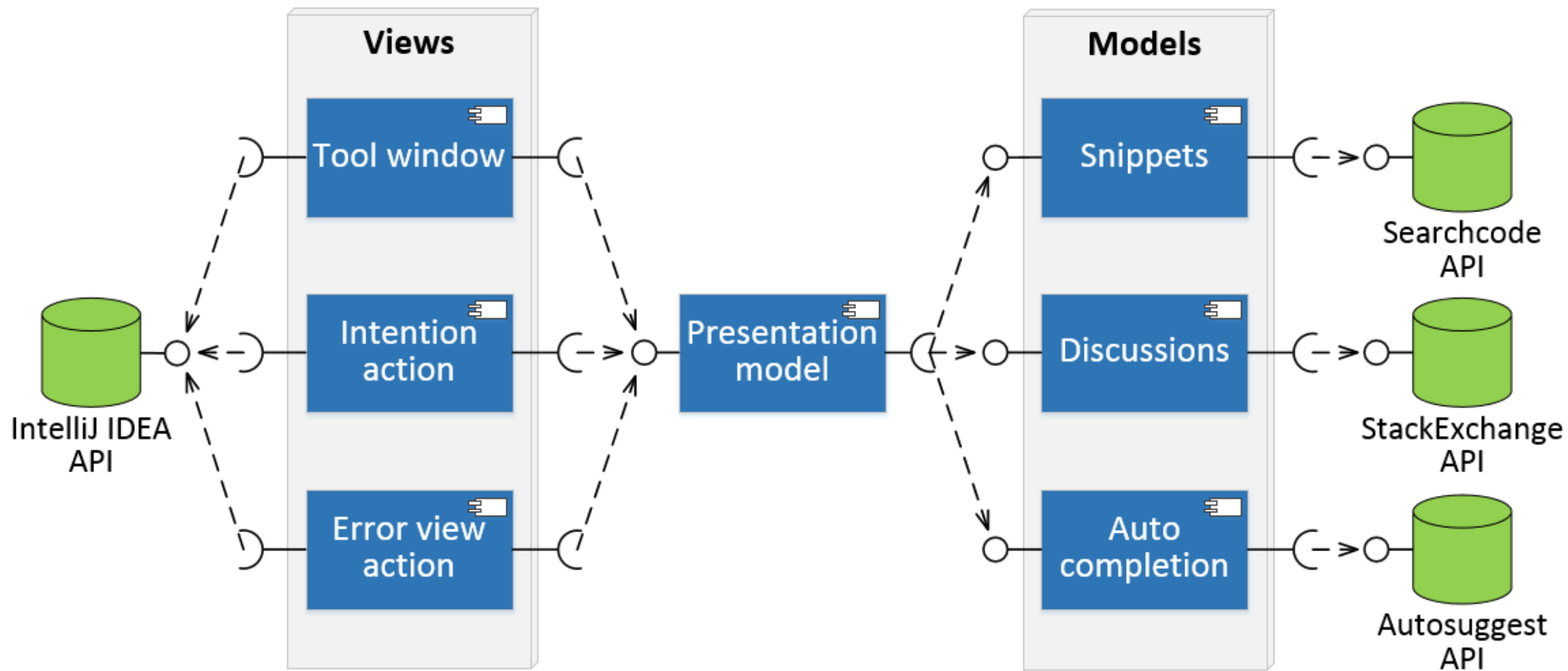
- Источник данных: сторонние сервисы
 - SearchCode API
 - Stack Exchange API
- Преимущества
 - потенциально выше точность рекомендации
 - не требуется сбор данных

Ранжирование фрагментов кода



- Ключевые слова – идентификаторы из кода пользователя
 - обработка выполняется полностью локально
- Векторная модель
 - вычисление веса ключевых слов с помощью TF–IDF
 - оценка схожести векторов через косинус угла между ними
- Подход не зависит от языка программирования

Архитектура



Пользовательский интерфейс



Assistant

convert string to int | in java | < > | ☆ 1/7

- convert string to int
- convert string to date
- convert string to array
- convert string to double
- convert string to long

```
1
2
3
4 * @param def default value
5 * @return
6 */
7 public static int toInt(String value, int def) {
8     if (isEmpty(value)) {
9         return def;
10    }
11    try {
12        return Integer.valueOf(value);
13    } catch (NumberFormatException e) {
14        e.printStackTrace();
15        return def;
16    }
17 }
```

Пользовательский интерфейс



The screenshot shows the Messages window in an IDE, titled "Build". It contains a list of error messages:

- Error:(22, 17) java: package org.junit does not exist
- Error:(24, 24) java: package org.junit does not exist
- Error:(24, 1) java: static import only from classes and interfaces
- Error:(25, 24) java: package
- Error:(25, 1) java: static imp
- Error:(29, 4) java: cannot fin
- symbol: clas
- location: class
- Error:(31, 5) java: cannot fin

A context menu is open over the error "Error:(25, 24) java: package", listing the following actions:

- Jump to Source F4
- Copy Ctrl+C
- Exclude from Compile
- Find more information (highlighted)
- Exclude from Validation

At the bottom of the Messages window, there are tabs for "0: Messages", "6: TODO", and "Terminal".

Апробация

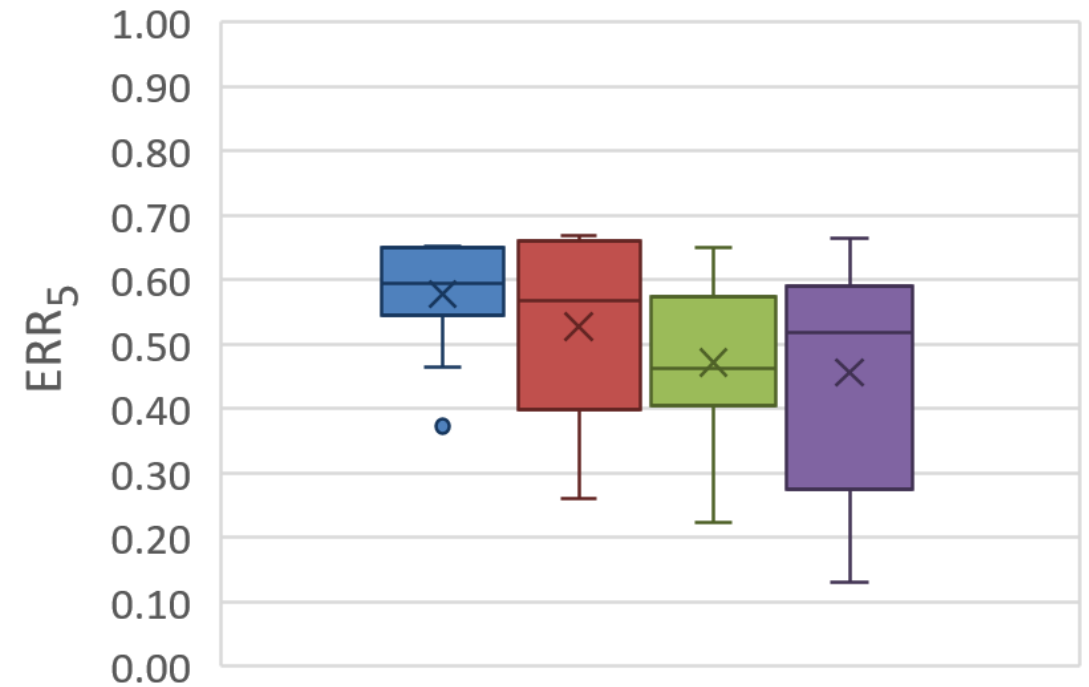
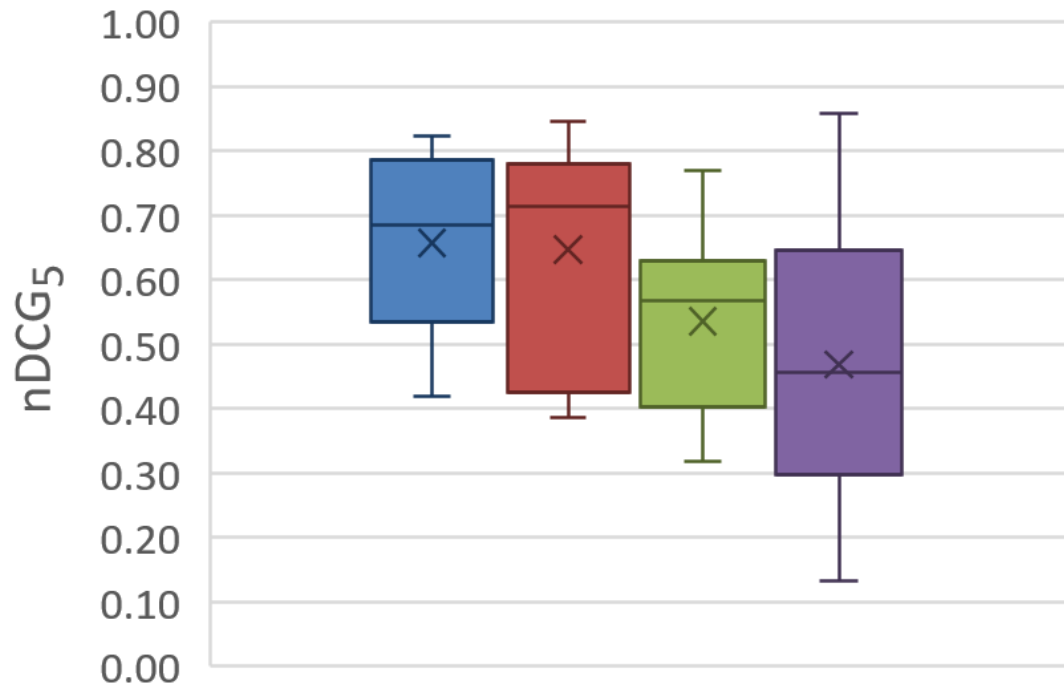


- Целевой язык – Java
- 10 запросов из списка наиболее часто задаваемых вопросов в Stack Overflow
- Данные: 5 первых фрагментов кода по каждому запросу
- Экспертная оценка

Апробация



■ Разработанное решение ■ NLP2Code ■ Codota ■ Поиск без ранжирования



$$nDCG_K = \frac{DCG_K}{IdealDCG_K} \quad DCG_K = \sum_{k=1}^K \frac{2^{rel(k)} - 1}{\log_2(k + 1)}$$

$$ERR_K = \frac{1}{K} \sum_{k=1}^K \frac{1}{k} \Pr(\text{пользователь остановится на } k - \text{м фрагменте})$$

Результаты



- Сделан обзор существующих решений
- Реализован алгоритм рекомендации
- Разработана и реализована архитектура расширения
- Реализован пользовательский интерфейс
- Проведено сравнение инструмента с аналогами