全国软件与应用学术会议（NASAC 2019）

分领域：新教师论坛

面向API文档的数据分析研究

张静宣

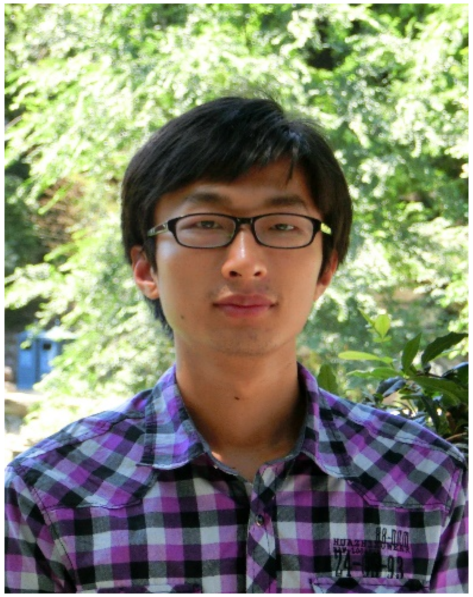
2019年11月24日

南京航空航天大学
Nanjing University of Aeronautics and Astronautics

自我介绍

张静宣(Jingxuan Zhang)

助理教授，硕士生导师
南京航空航天大学 计算机科学与技术学院
Email: jxzhang@nuaa.edu.cn
主页: https://csjxzhang.github.io
研究方向：软件数据分析，软件仓库挖掘



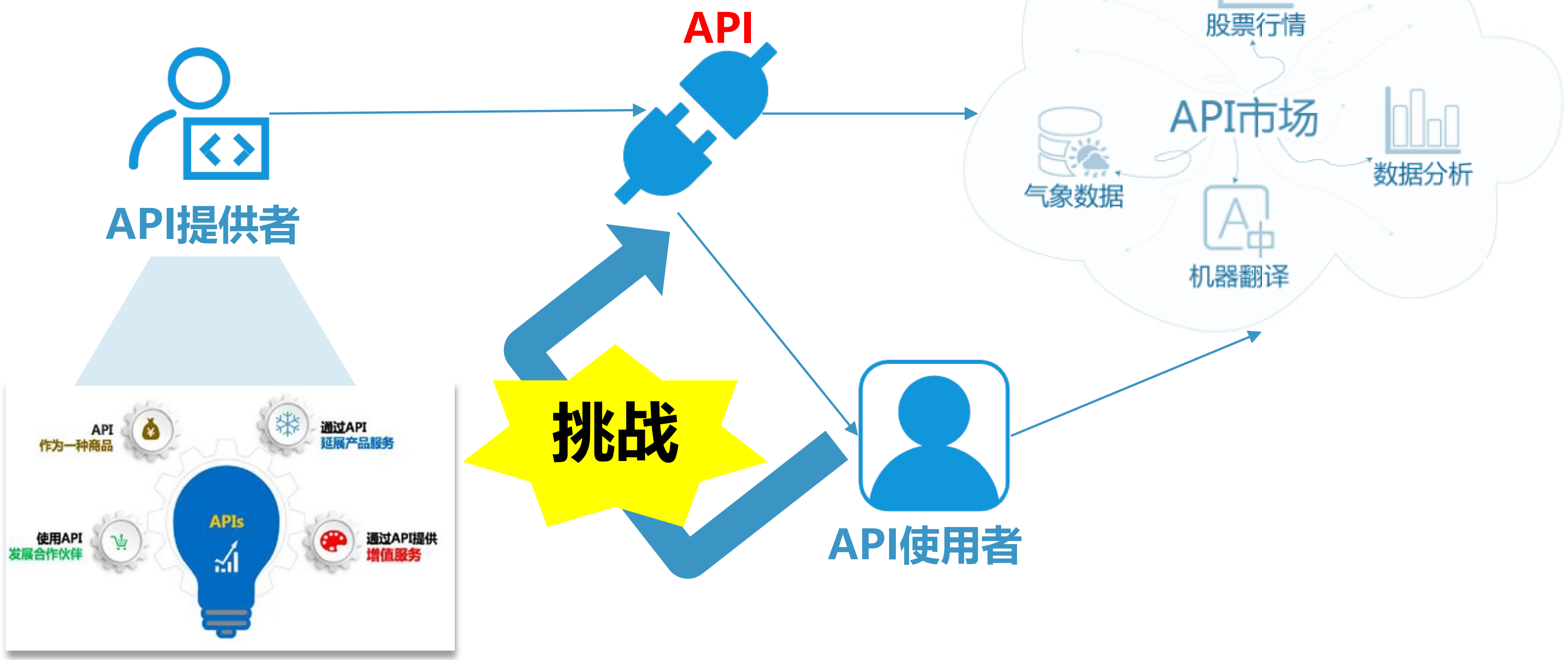
时间	单位	经历	导师
2008.9-2012.6	大连理工大学	本科	--
2012.9-2018.6	大连理工大学	博士	江贺
2014.5-2014.8	新加坡管理大学	研究助理	David Lo
2018.7-现在	南京航空航天大学	助理教授	--

汇报提纲

API和API文档

南京航空航天大学
Nanjing University of Aeronautics and Astronautics

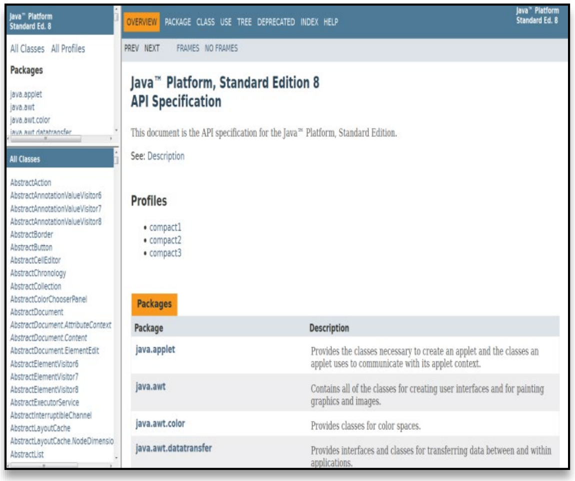
API衍生新的商业模式



API文档-帮助开发者学习API的正确使用

API文档的分类

南京航空航天大学 Nanjing University of Aeronautics and Astronautics



Java™ Platform Standard Ed. 8

All Classes All Profiles

Packages

java.applet
java.awt
java.awt.color
java.awt.datatransfer

All Classes

AbstractAction
AbstractAnnotationValueVisitor6
AbstractAnnotationValueVisitor7
AbstractAnnotationValueVisitor8
AbstractBorder
AbstractButton
AbstractCellEditor
AbstractChronology
AbstractCollection
AbstractColorChooserPanel
AbstractDocument
AbstractDocument.AttributeContext
AbstractDocument.Content
AbstractDocument.ElementEdit
AbstractElementVisitor6
AbstractElementVisitor7
AbstractElementVisitor8
AbstractExecutorService
AbstractInterruptibleChannel
AbstractLayoutCache
AbstractLayoutCache.NodeDimensions
AbstractList

OVERVIEW PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

PREV NEXT FRAMES NO FRAMES

Java™ Platform, Standard Edition 8
API Specification

This document is the API specification for the Java™ Platform, Standard Edition.

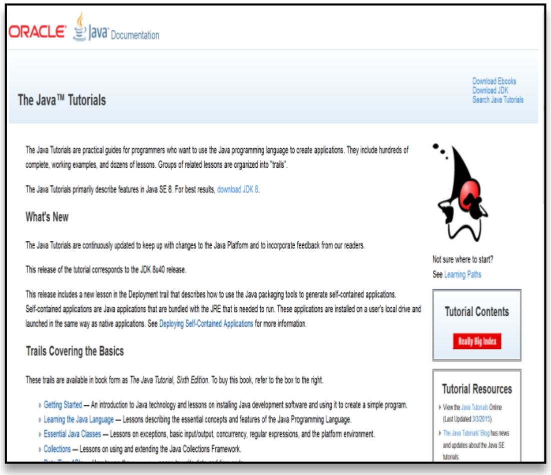
See: Description

Profiles

- compact1
- compact2
- compact3

Packages

Package	Description
java.applet	Provides the classes necessary to create an applet and the classes an applet uses to communicate with its applet context.
java.awt	Contains all of the classes for creating user interfaces and for painting graphics and images.
java.awt.color	Provides classes for color spaces.
java.awt.datatransfer	Provides interfaces and classes for transferring data between and within applications.



ORACLE Java Documentation

The Java™ Tutorials

Download Ebooks
Download JDK
Search Java Tutorials

The Java Tutorials are practical guides for programmers who want to use the Java programming language to create applications. They include hundreds of complete, working examples, and dozens of lessons. Groups of related lessons are organized into "trails".

The Java Tutorials primarily describe features in Java SE 8. For best results, download JDK 8.

What's New

The Java Tutorials are continuously updated to keep up with changes to the Java Platform and to incorporate feedback from our readers.

This release of the tutorial corresponds to the JDK 8u40 release.

This release includes a new lesson in the Deployment trail that describes how to use the Java packaging tools to generate self-contained applications. Self-contained applications are Java applications that are bundled with the JRE that is needed to run. These applications are installed on a user's local drive and launched in the same way as native applications. See Deploying Self-Contained Applications for more information.

Trails Covering the Basics

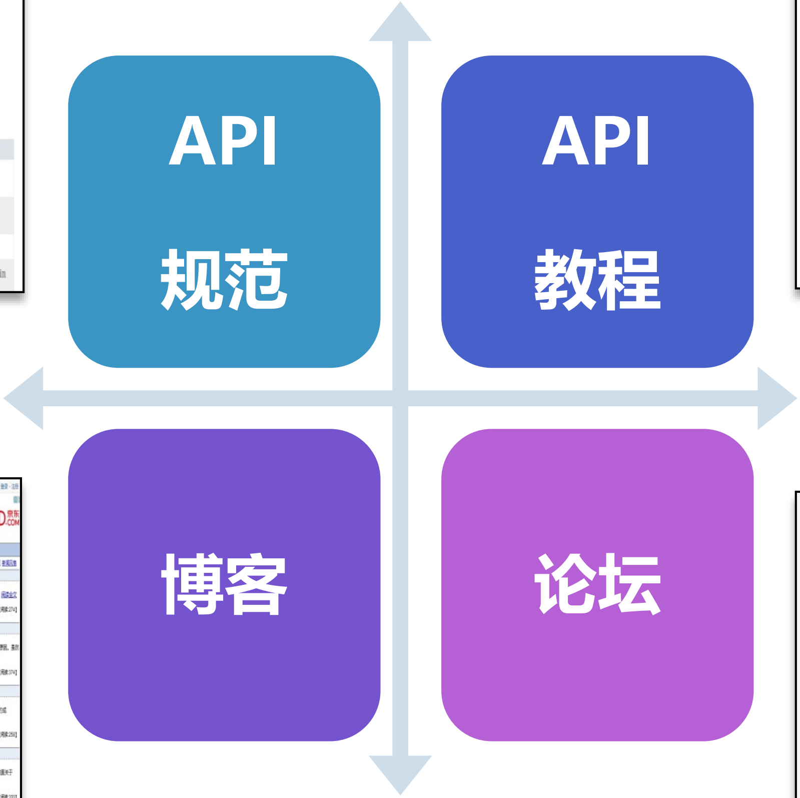
These trails are available in book form as The Java Tutorial, Sixth Edition. To buy this book, refer to the box to the right.

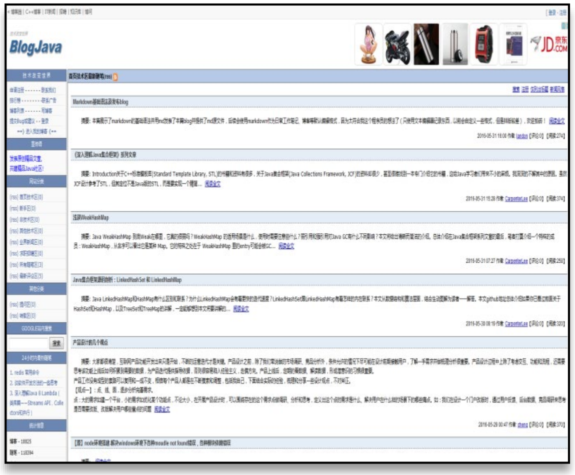
- Getting Started — An introduction to Java technology and lessons on installing Java development software and using it to create a simple program.
- Learning the Java Language — Lessons describing the essential concepts and features of the Java Programming Language.
- Essential Java Classes — Lessons on exceptions, basic input/output, concurrency, regular expressions, and the platform environment.
- Collections — Lessons on using and extending the Java Collections Framework.

Not sure where to start?
See Learning Paths

Tutorial Contents
Really Big Index

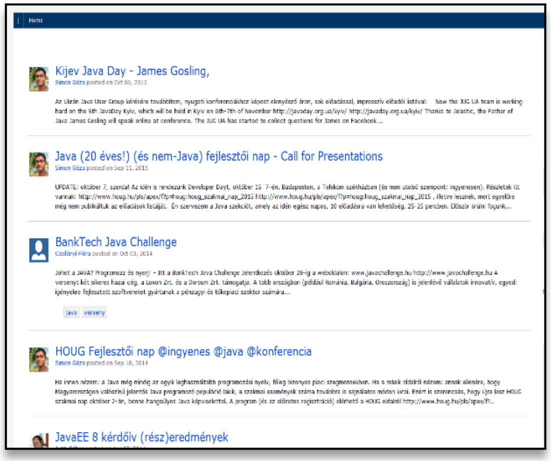
Tutorial Resources
View the Java Tutorials Online (Last Updated 3/3/2015)
The Java Tutorials' Blog has news and updates about the Java SE tutorials.





BlogJava

[illegible]



Home

Kijev Java Day - James Gosling,
Simon Géza posted on Oct 05, 2015

Az Ukrán Java User Group körében továbbítom, nyugati konferenciákhoz képest olcsóbb áron, sok előadással, ingyenes előadói látóval: Now the JUG UA team is working hard on the 5th JavaDay Kyiv, which will be held in Kyiv on 6th-7th of November http://javaday.org.ua/kyiv/ http://javaday.org.ua/kyiv/ Thanks to Jelastic, the Father of Java James Gosling will speak online at conference. The JUG UA has started to collect questions for James on Facebook...

Java (20 éves!) (és nem-Java) fejlesztői nap - Call for Presentations
Simon Géza posted on Sep 11, 2015

UPDATE: október 7, szerda! Az idén is rendezünk Developer Dayt, október 15 7-én, Budapesten, a Telekom székházban (és nem utolsó szempont: ingyenesen). Részletek itt várnak: http://www.houg.hu/pls/apex/f?p=houg:houg_szakmai_nap_2015 ... Ebben a szerveztem a Java szekciót, amely az idén egész napos, 10 előadásra van lehetőség, 25-25 percben. Előzör önálló fogunk...

BankTech Java Challenge
Csatári Flóra posted on Oct 03, 2014

Jóhet a JAVA? Programozz és nyerj! - Itt a BankTech Java Challenge Jelentkezés október 28-ig a weboldalon: www.javachallenge.hu http://www.javachallenge.hu A versenyt két síkeres hazai cég, a Loxon Zrt. és a Dorsum Zrt. támogatja. A több országban (például Románia, Bulgária, Oroszország) is jelenlévő vállalatok innovatív, egyedi igényekre fejlesztett szoftvereket gyártanak a pénzügyi és tőkepiaci szektor számára...

java verseny

HOUG Fejlesztői nap @ingyenes @java @konferencia
Simon Géza posted on Sep 18, 2014

Ha innen nézem: a Java még mindig az egyik leghasználtabb programozási nyelv, illetve bizonyos piaci szegmensekben. Ha a másik oldalról nézem: ... http://www.houg.hu/pls/apex/f?...

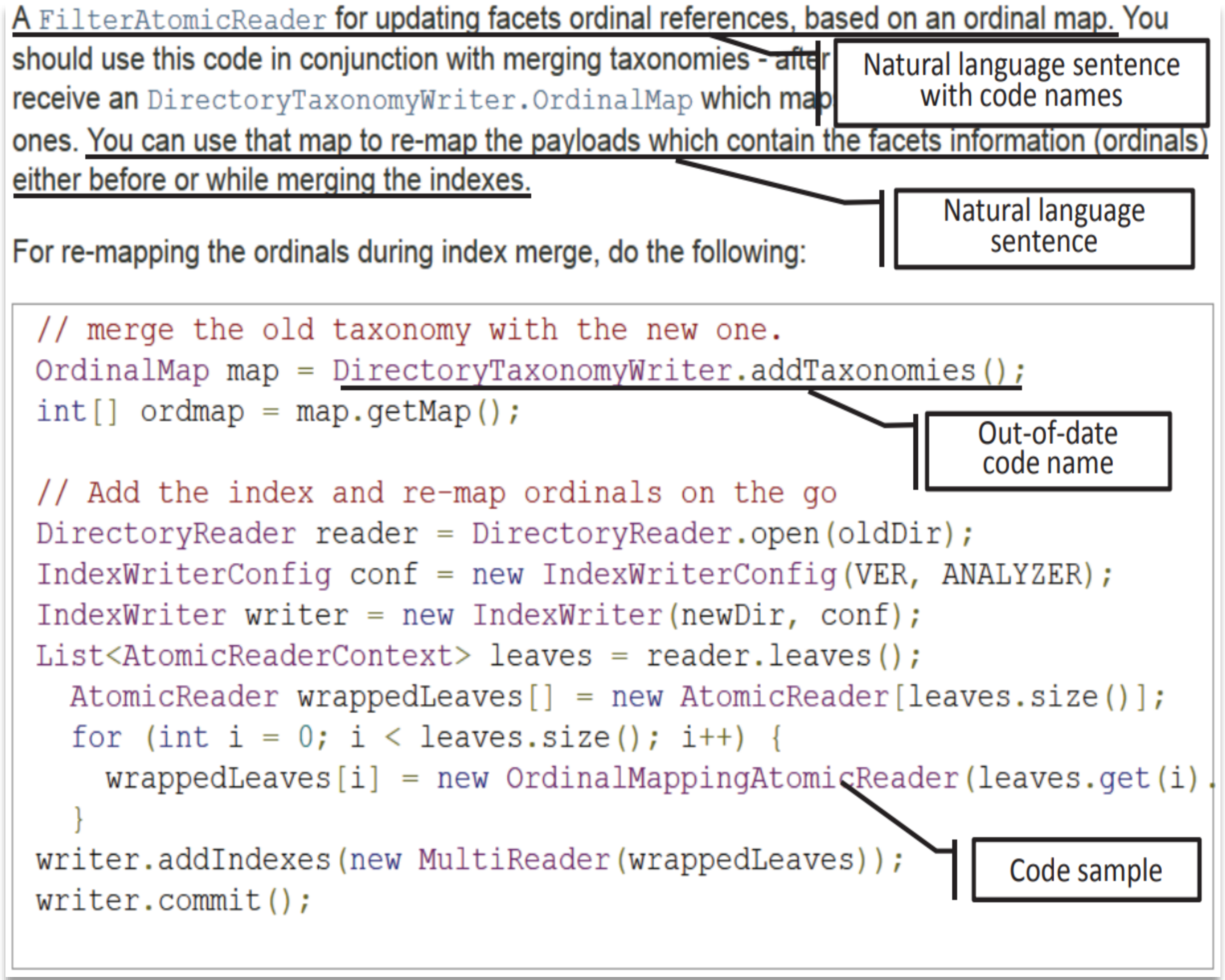
JavaEE 8 kérdőív (rész)eredmények

Maalej W, Robillard M P. Patterns of knowledge in API reference documentation, IEEE Transactions on Software Engineering, 2013, 39(9): 1264-1282.

API文档的内容

南京航空航天大学 Nanjing University of Aeronautics and Astronautics

既包含有自然语言的描述，又包含少量代码样例。同时自然语言的描述有可能也包含代码元素。



A FilterAtomicReader for updating facets ordinal references, based on an ordinal map. You should use this code in conjunction with merging taxonomies - after receive an DirectoryTaxonomyWriter.OrdinalMap which map ones. You can use that map to re-map the payloads which contain the facets information (ordinals) either before or while merging the indexes.

For re-mapping the ordinals during index merge, do the following:

```
// merge the old taxonomy with the new one.
OrdinalMap map = DirectoryTaxonomyWriter.addTaxonomies();
int[] ordmap = map.getMap();

// Add the index and re-map ordinals on the go
DirectoryReader reader = DirectoryReader.open(oldDir);
IndexWriterConfig conf = new IndexWriterConfig(VER, ANALYZER);
IndexWriter writer = new IndexWriter(newDir, conf);
List<AtomicReaderContext> leaves = reader.leaves();
  AtomicReader wrappedLeaves[] = new AtomicReader[leaves.size()];
  for (int i = 0; i < leaves.size(); i++) {
    wrappedLeaves[i] = new OrdinalMappingAtomicReader(leaves.get(i).
  }
writer.addIndexes(new MultiReader(wrappedLeaves));
writer.commit();
```

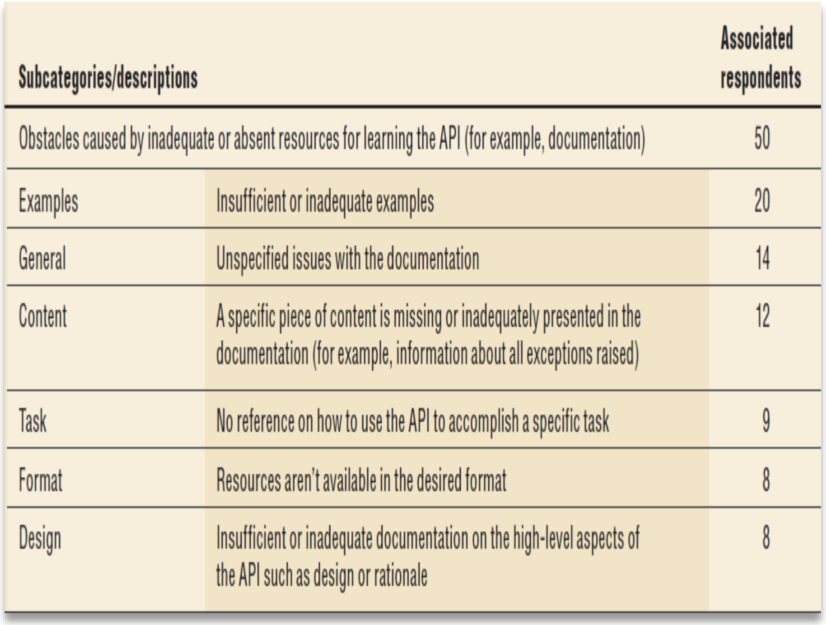
Natural language sentence with code names
Natural language sentence
Out-of-date code name
Code sample

Zhong H, Su Z. Detecting API documentation errors, International Conference on Object Oriented Programming Systems Languages & Applications (OOPSLA). ACM, 2013:803-816.

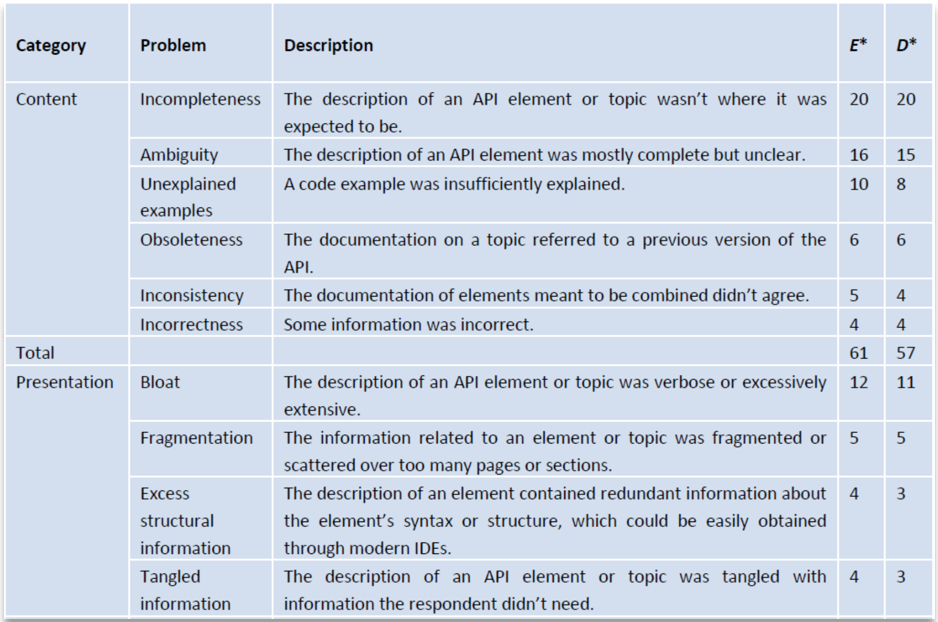
API文档的质量

南京航空航天大学 Nanjing University of Aeronautics and Astronautics

研究者对API文档的质量做了实证研究，调研API学习的障碍和API文档失效的原因。



Subcategories/descriptions		Associated respondents
Obstacles caused by inadequate or absent resources for learning the API (for example, documentation)		50
Examples	Insufficient or inadequate examples	20
General	Unspecified issues with the documentation	14
Content	A specific piece of content is missing or inadequately presented in the documentation (for example, information about all exceptions raised)	12
Task	No reference on how to use the API to accomplish a specific task	9
Format	Resources aren't available in the desired format	8
Design	Insufficient or inadequate documentation on the high-level aspects of the API such as design or rationale	8



Category	Problem	Description	*E**	*D**
Content	Incompleteness	The description of an API element or topic wasn't where it was expected to be.	20	20
	Ambiguity	The description of an API element was mostly complete but unclear.	16	15
	Unexplained examples	A code example was insufficiently explained.	10	8
	Obsoleteness	The documentation on a topic referred to a previous version of the API.	6	6
	Inconsistency	The documentation of elements meant to be combined didn't agree.	5	4
	Incorrectness	Some information was incorrect.	4	4
Total			61	57
Presentation	Bloat	The description of an API element or topic was verbose or excessively extensive.	12	11
	Fragmentation	The information related to an element or topic was fragmented or scattered over too many pages or sections.	5	5
	Excess structural information	The description of an element contained redundant information about the element's syntax or structure, which could be easily obtained through modern IDEs.	4	3
	Tangled information	The description of an API element or topic was tangled with information the respondent didn't need.	4	3

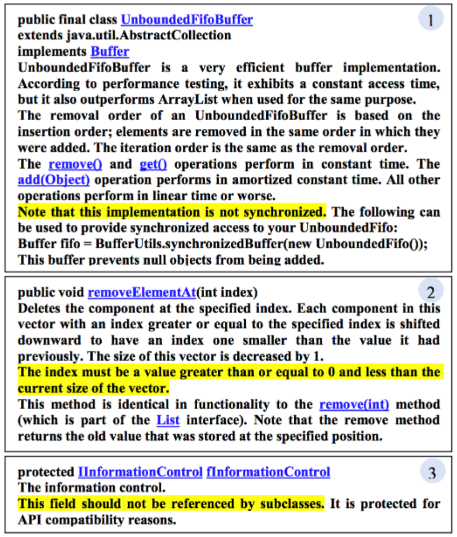
质量问题		
重要知识不凸显	➡	自动识别并高亮重要知识
重要信息（代码）缺失	➡	自动填充缺失的重要信息
信息冗长且碎片化	➡	自动切分并推荐相关内容

Uddin G, Robillard M P. How API Documentation Fails[J]. IEEE Software, 2015, 32(4):68-75.
Robillard M P. What Makes APIs Hard to Learn? Answers from Developers[J]. IEEE Software, 2009, 26(6):27-34.

研究问题

南京航空航天大学
Nanjing University of Aeronautics and Astronautics

API文档 内容解析

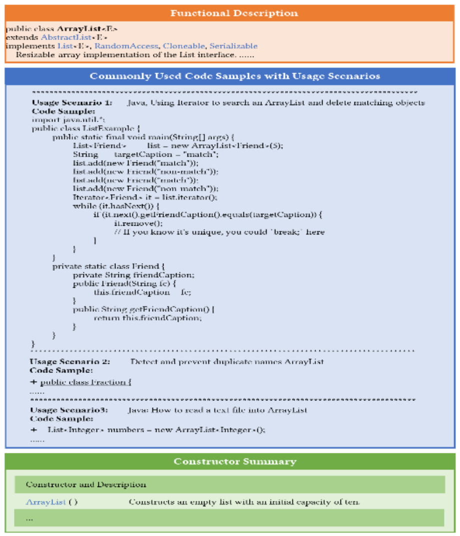

public final class UnboundedFifoBuffer
extends java.util.AbstractCollection
implements Buffer
UnboundedFifoBuffer is a very efficient buffer implementation. According to performance testing, it exhibits a constant access time, but it also outperforms ArrayList when used for the same purpose. The removal order of an UnboundedFifoBuffer is based on the insertion order; elements are removed in the same order in which they were added. The iteration order is the same as the removal order.
The remove() and get() operations perform in constant time. The add(Object) operation performs in amortized constant time. All other operations perform in linear time or worse.
Note that this implementation is not synchronized. The following can be used to provide synchronized access to your UnboundedFifo:
Buffer fifo = BufferUtils.synchronizedBuffer(new UnboundedFifo());
This buffer prevents null objects from being added. 1

public void removeElementAt(int index) 2
Deletes the component at the specified index. Each component in this vector with an index greater or equal to the specified index is shifted downward to have an index one smaller than the value it had previously. The size of this vector is decreased by 1.
The index must be a value greater than or equal to 0 and less than the current size of the vector.
This method is identical in functionality to the remove(int) method (which is part of the List interface). Note that the remove method returns the old value that was stored at the specified position.

protected IInformationControl fInformationControl 3
The information control.
This field should not be referenced by subclasses. It is protected for API compatibility reasons.

解决重要知识不凸显的问题

API文档 信息增强


Functional Description

public class **ArrayList<E>**
extends AbstractList<E>
implements List<E>, RandomAccess, Cloneable, Serializable
Resizable array implementation of the List interface. ……

Commonly Used Code Samples with Usage Scenarios

Usage Scenario 1: Java, Using Iterator to search an ArrayList and delete matching objects
Code Sample:

```
import java.util.*;
public class ListExample {
    public static final void main(String[] args) {
        List<Friend>      list = new ArrayList<Friend>(5);
        String      targetCaption = "match";
        list.add(new Friend("match"));
        list.add(new Friend("non-match"));
        list.add(new Friend("match"));
        list.add(new Friend("non match"));
        Iterator<Friend> it = list.iterator();
        while (it.hasNext()) {
            if (it.next().getFriendCaption().equals(targetCaption)) {
                it.remove();
                // If you know it's unique, you could `break;` here
            }
        }
    }
    private static class Friend {
        private String friendCaption;
        public Friend(String fc) {
            this.friendCaption    fc;
        }
        public String getFriendCaption() {
            return this.friendCaption;
        }
    }
}
```

Usage Scenario 2: Detect and prevent duplicate names ArrayList
Code Sample:
+ public class Fraction {
……

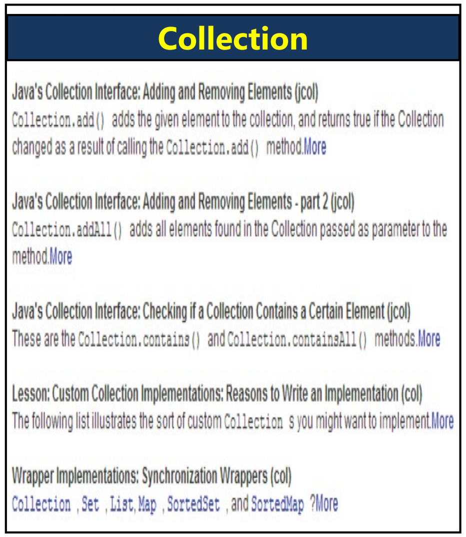
Usage Scenario3: Java: How to read a text file into ArrayList
Code Sample:
+ List<Integer> numbers = new ArrayList<Integer>();
……

Constructor Summary

Constructor and Description	
ArrayList ( )	Constructs an empty list with an initial capacity of ten.
...	

解决重要信息（代码）缺失的问题

API文档 参考推荐


Collection

Java's Collection Interface: Adding and Removing Elements (jcol)
Collection.add() adds the given element to the collection, and returns true if the Collection changed as a result of calling the Collection.add() method.More

Java's Collection Interface: Adding and Removing Elements - part 2 (jcol)
Collection.addAll() adds all elements found in the Collection passed as parameter to the method.More

Java's Collection Interface: Checking if a Collection Contains a Certain Element (jcol)
These are the Collection.contains() and Collection.containsAll() methods.More

Lesson: Custom Collection Implementations: Reasons to Write an Implementation (col)
The following list illustrates the sort of custom Collection s you might want to implement.More

Wrapper Implementations: Synchronization Wrappers (col)
Collection, Set, List, Map, SortedSet, and SortedMap ?More

解决信息冗长且碎片化问题

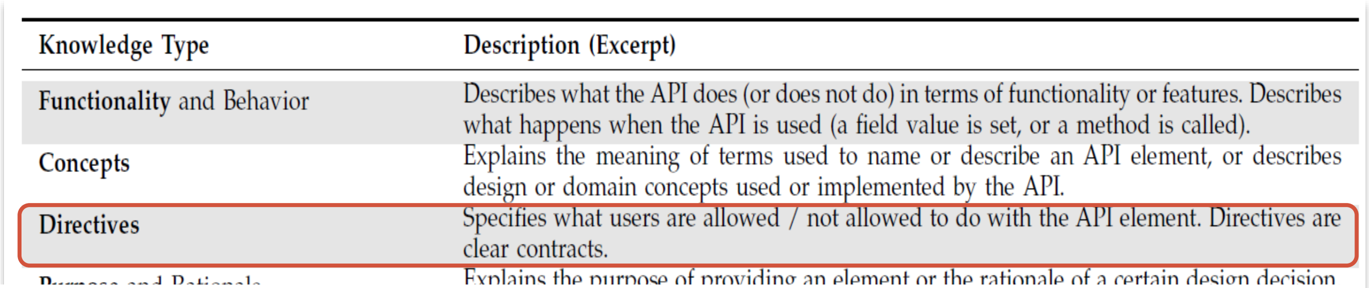
汇报提纲

南京航空航天大学
Nanjing University of Aeronautics and Astronautics

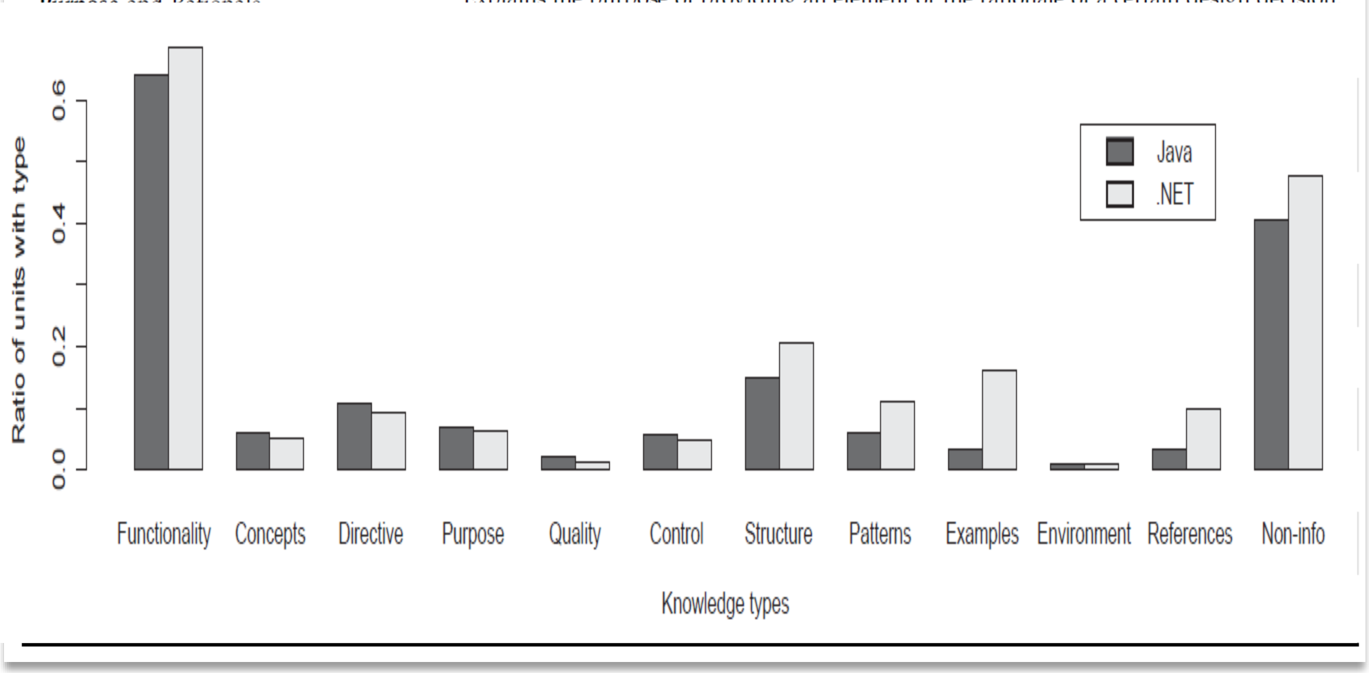
API文档内容解析

南京航空航天大学 Nanjing University of Aeronautics and Astronautics

已有研究者对API文档包含的知识进行了分析，发现了12种知识类型以及他们的出现模式。



Knowledge Type	Description (Excerpt)
Functionality and Behavior	Describes what the API does (or does not do) in terms of functionality or features. Describes what happens when the API is used (a field value is set, or a method is called).
Concepts	Explains the meaning of terms used to name or describe an API element, or describes design or domain concepts used or implemented by the API.
Directives	Specifies what users are allowed / not allowed to do with the API element. Directives are clear contracts.
Purpose and Rationale	Explains the purpose of providing an element or the rationale of a certain design decision



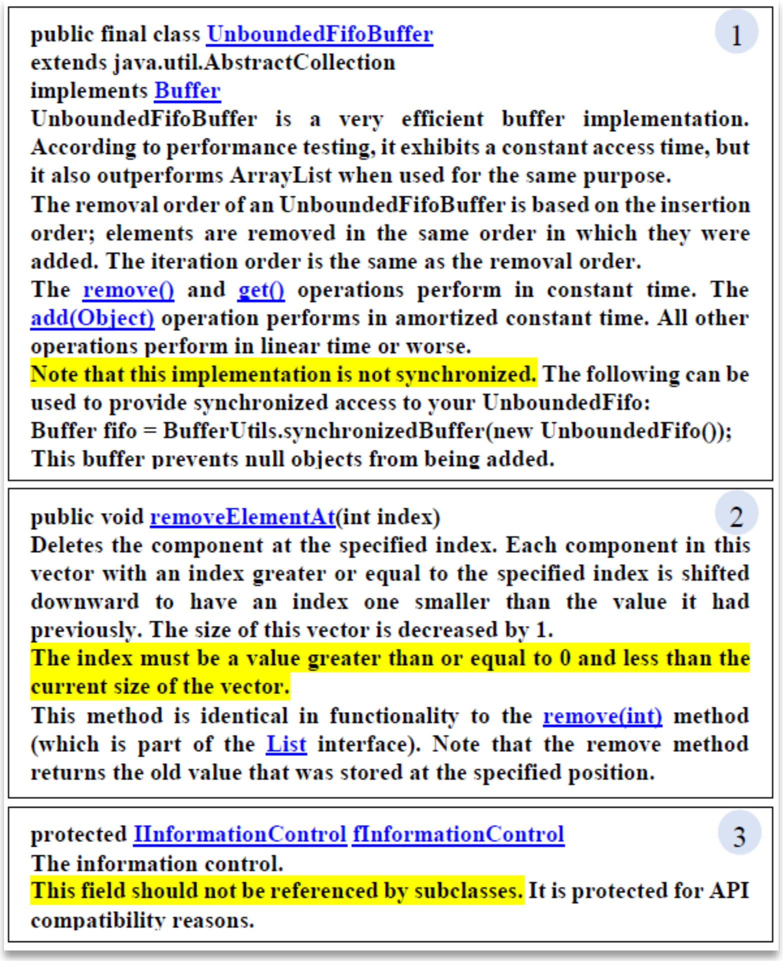
Maalej W, Robillard M P. Patterns of knowledge in API reference documentation, IEEE Transactions on Software Engineering, 2013, 39(9): 1264-1282.

API文档内容解析

南京航空航天大学
Nanjing University of Aeronautics and Astronautics

定义： API directive是正确调用API所必须遵守的约束和限制

目标： 自动的识别和高亮API文档中的directive，提醒开发者



1

public final class UnboundedFifoBuffer
extends java.util.AbstractCollection
implements Buffer
UnboundedFifoBuffer is a very efficient buffer implementation. According to performance testing, it exhibits a constant access time, but it also outperforms ArrayList when used for the same purpose.
The removal order of an UnboundedFifoBuffer is based on the insertion order; elements are removed in the same order in which they were added. The iteration order is the same as the removal order.
The remove() and get() operations perform in constant time. The add(Object) operation performs in amortized constant time. All other operations perform in linear time or worse.
Note that this implementation is not synchronized. The following can be used to provide synchronized access to your UnboundedFifo:
Buffer fifo = BufferUtils.synchronizedBuffer(new UnboundedFifo());
This buffer prevents null objects from being added.

2

public void removeElementAt(int index)
Deletes the component at the specified index. Each component in this vector with an index greater or equal to the specified index is shifted downward to have an index one smaller than the value it had previously. The size of this vector is decreased by 1.
The index must be a value greater than or equal to 0 and less than the current size of the vector.
This method is identical in functionality to the remove(int) method (which is part of the List interface). Note that the remove method returns the old value that was stored at the specified position.

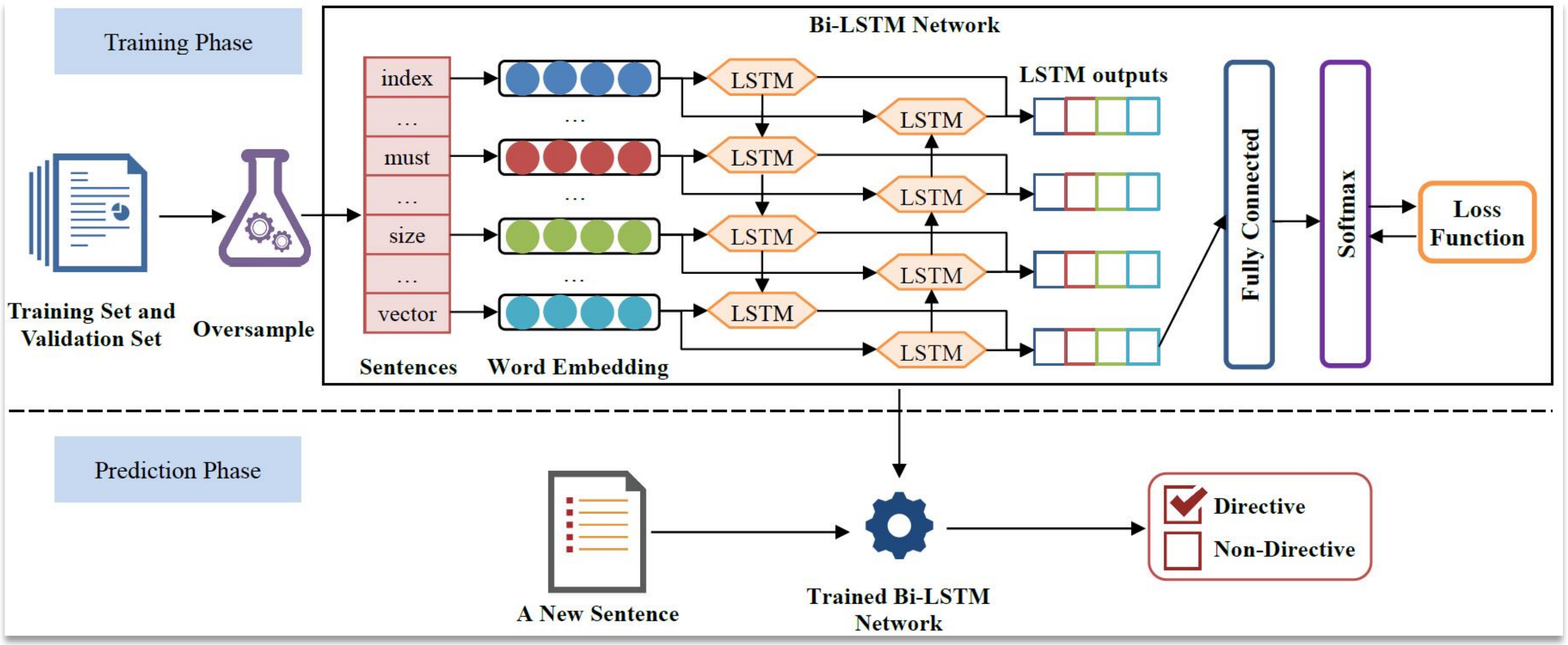
3

protected IInformationControl fInformationControl
The information control.
This field should not be referenced by subclasses. It is protected for API compatibility reasons.

API文档内容解析

南京航空航天大学 Nanjing University of Aeronautics and Astronautics

解决方案: 带采样的深度学习（LSTM）模型



Jingxuan Zhang, He Jiang, Shuai Lu, Ge Li, and Xin Chen. DeepDir: A Deep Learning Approach for API Directive Detection, *SCIENCE CHINA Information Sciences*. 2019, Accepted.

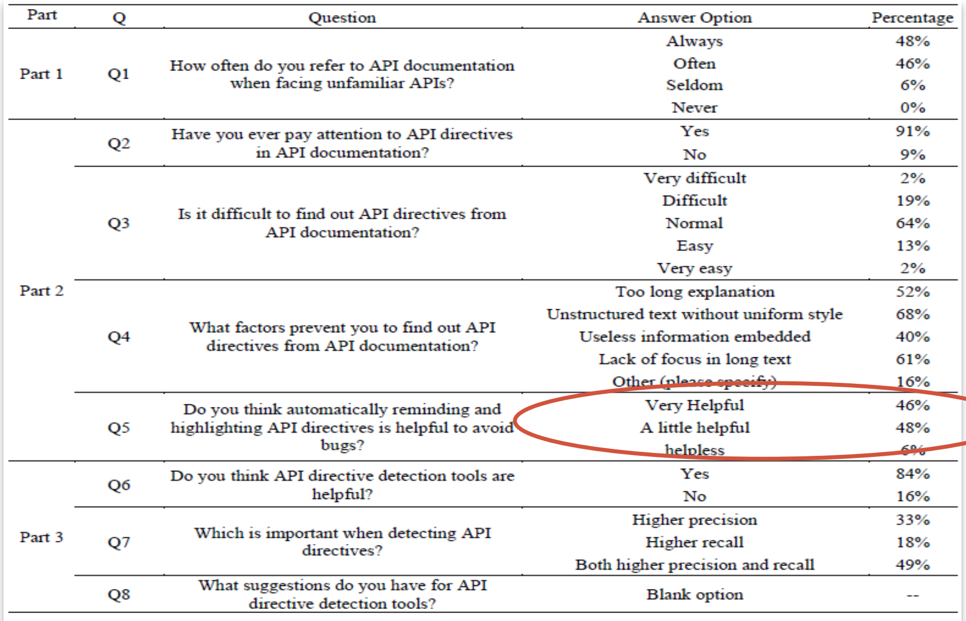
API文档内容解析

南京航空航天大学
Nanjing University of Aeronautics and Astronautics

1. 我们提出的方法是否超过对比方法?

API Specification	Precision			Recall			F-Measure		
	Baseline	DeepDir	Improv.	Baseline	DeepDir	Improv.	Baseline	DeepDir	Improv.
Java	15.18%	34.43%	+19.25%	73.81%	76.17%	+2.36%	25.18%	47.39%	+22.21%
JFace	28.75%	54.61%	+25.86%	78.40%	81.09%	+2.69%	42.07%	64.90%	+22.83%
commons.collections	49.37%	65.82%	+16.45%	67.41%	85.44%	+18.03%	57.00%	74.30%	+17.30%
Average	31.10%	51.62%	+20.52%	73.21%	80.90%	+7.69%	41.42%	62.20%	+20.78%

2. 自动的识别高亮directive是否对开发者有利?



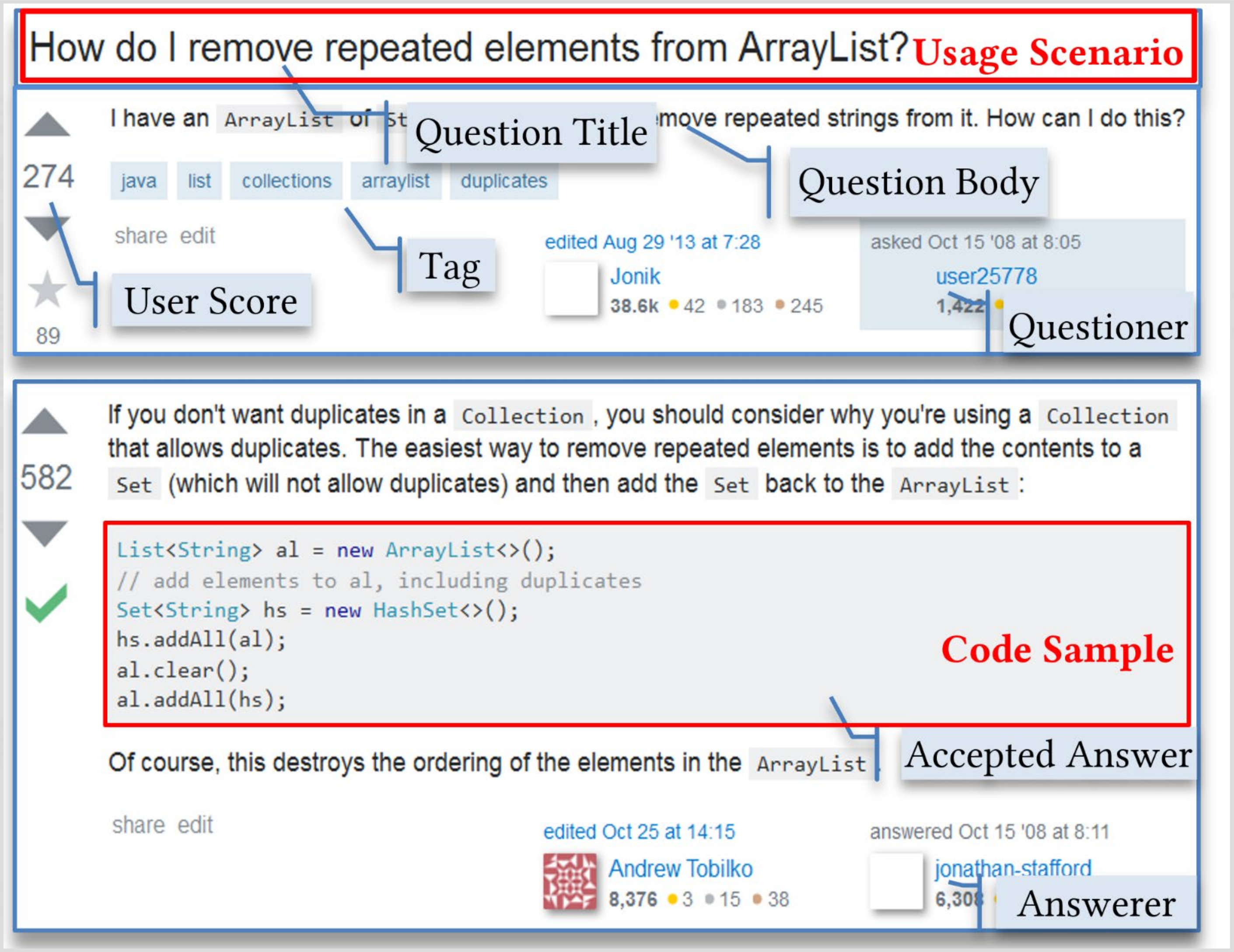
Part	Q	Question	Answer Option	Percentage
Part 1	Q1	How often do you refer to API documentation when facing unfamiliar APIs?	Always	48%
			Often	46%
			Seldom	6%
			Never	0%
Part 2	Q2	Have you ever pay attention to API directives in API documentation?	Yes	91%
			No	9%
	Q3	Is it difficult to find out API directives from API documentation?	Very difficult	2%
			Difficult	19%
			Normal	64%
			Easy	13%
			Very easy	2%
	Q4	What factors prevent you to find out API directives from API documentation?	Too long explanation	52%
			Unstructured text without uniform style	68%
			Useless information embedded	40%
			Lack of focus in long text	61%
			Other (please specify)	16%
	Q5	Do you think automatically reminding and highlighting API directives is helpful to avoid bugs?	Very Helpful	46%
			A little helpful	48%
			helpless	6%
Part 3	Q6	Do you think API directive detection tools are helpful?	Yes	84%
			No	16%
	Q7	Which is important when detecting API directives?	Higher precision	33%
			Higher recall	18%
			Both higher precision and recall	49%
	Q8	What suggestions do you have for API directive detection tools?	Blank option	--

汇报提纲

南京航空航天大学
Nanjing University of Aeronautics and Astronautics

API文档信息增强

南京航空航天大学
Nanjing University of Aeronautics and Astronautics


How do I remove repeated elements from ArrayList?

I have an ArrayList of St [illegible] move repeated strings from it. How can I do this?

java list collections arraylist duplicates

274

share edit

edited Aug 29 '13 at 7:28
Jonik
38.6k ● 42 ● 183 ● 245

asked Oct 15 '08 at 8:05
user25778
1,422

89

582

If you don't want duplicates in a `Collection`, you should consider why you're using a `Collection` that allows duplicates. The easiest way to remove repeated elements is to add the contents to a `Set` (which will not allow duplicates) and then add the `Set` back to the `ArrayList`:

```
List<String> al = new ArrayList<>();
// add elements to al, including duplicates
Set<String> hs = new HashSet<>();
hs.addAll(al);
al.clear();
al.addAll(hs);
```

Of course, this destroys the ordering of the elements in the `ArrayList`.

share edit

edited Oct 25 at 14:15
Andrew Tobilko
8,376 ● 3 ● 15 ● 38

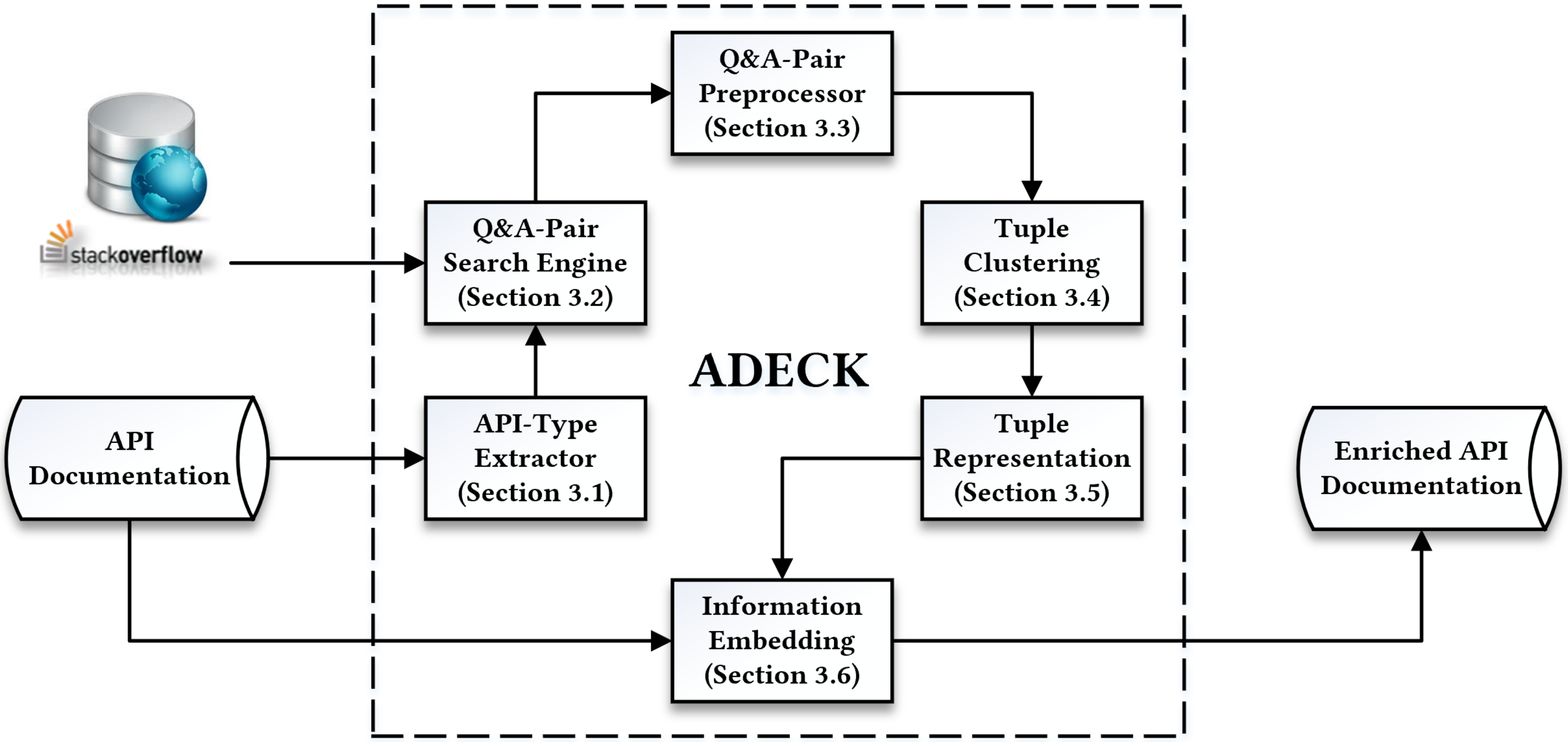
answered Oct 15 '08 at 8:11
jonathan-stafford
6,308

Kim J, Lee S, Hwang S W, et al. Enriching Documents with Examples: A Corpus Mining Approach, ACM Transactions on Information Systems, 2013, 31(1):157-160.

API文档信息增强

南京航空航天大学
Nanjing University of Aeronautics and Astronautics

解决方案：基于群智的代码样例填充方法



stackoverflow
Q&A-Pair Preprocessor (Section 3.3)
Q&A-Pair Search Engine (Section 3.2)
Tuple Clustering (Section 3.4)
ADECK
API Documentation
API-Type Extractor (Section 3.1)
Tuple Representation (Section 3.5)
Enriched API Documentation
Information Embedding (Section 3.6)

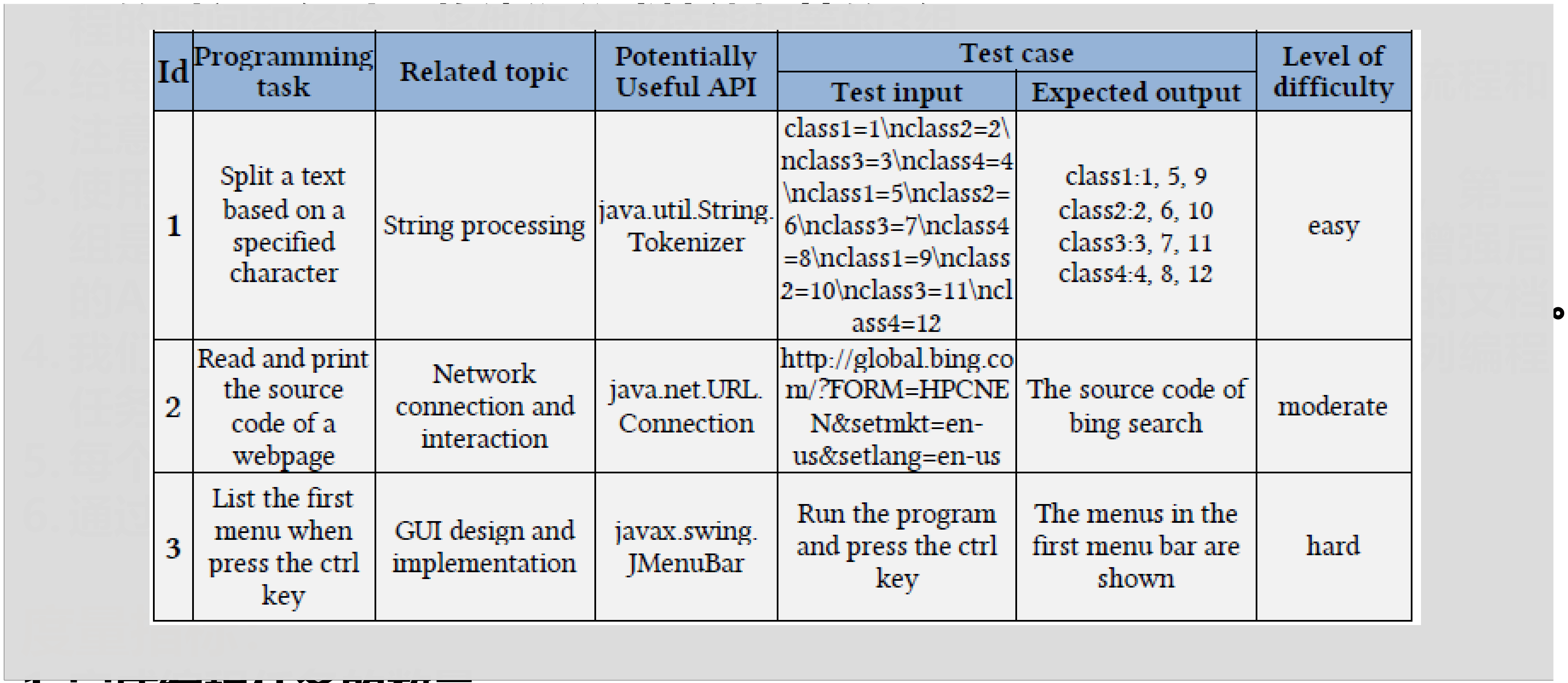
Jingxuan Zhang, He Jiang, Zhilei Ren, Tao Zhang, and Zhiqiu Huang. Enriching API Documentation with Code Samples and Usage Scenarios from Crowd Knowledge, *IEEE Transactions on Software Engineering (TSE)*. 2019, Accepted.

API文档信息增强

我们方法增强的API文档能否提高开发者效率?

方法步骤：

1.邀请21个参与者，设计问卷调查调研他们的编程背景。根据他们的学习编



Id	Programming task	Related topic	Potentially Useful API	Test case		Level of difficulty
				Test input	Expected output	
1	Split a text based on a specified character	String processing	java.util.String.Tokenizer	class1=1\nclass2=2\nclass3=3\nclass4=4\nclass1=5\nclass2=6\nclass3=7\nclass4=8\nclass1=9\nclass2=10\nclass3=11\nclass4=12	class1:1, 5, 9 class2:2, 6, 10 class3:3, 7, 11 class4:4, 8, 12	easy
2	Read and print the source code of a webpage	Network connection and interaction	java.net.URL.Connection	http://global.bing.com/?FORM=HPCNEN&setmkt=en-us&setlang=en-us	The source code of bing search	moderate
3	List the first menu when press the ctrl key	GUI design and implementation	javax.swing.JMenuBar	Run the program and press the ctrl key	The menus in the first menu bar are shown	hard

1.完成编程任务的数量
2.完成编程任务的时间

API文档信息增强

Nanjing University of Aeronautics and Astronautics

我们方法增强的API文档能否提高开发者效率?

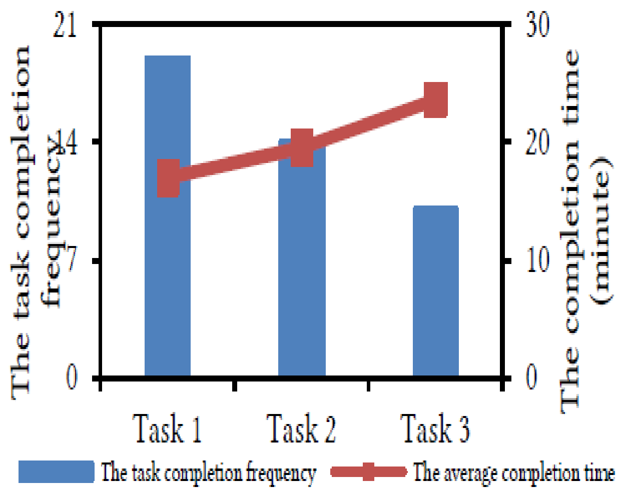


Fig. 8. Comparison between different tasks

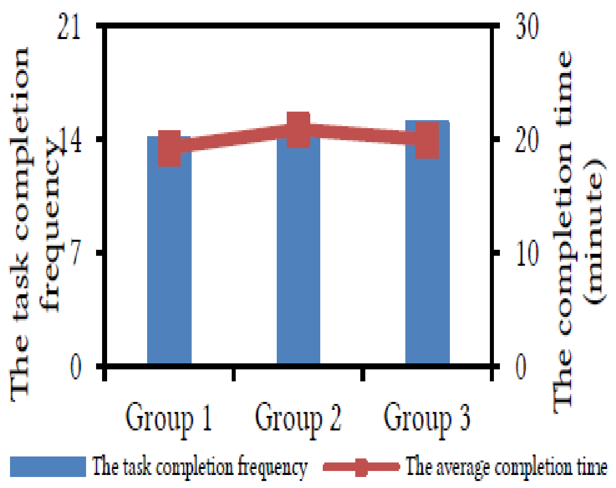


Fig. 9. Comparison between different groups

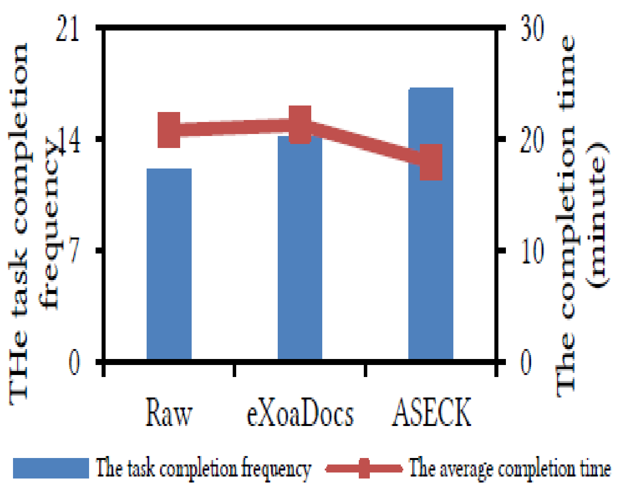


Fig. 10. Comparison between different API specifications

1. 编程问题设置的难易程度符合预期。
2. 我们的人员分组策略有效。
3. 利用我们方法增强的API文档，可以提高开发者的编程效率。

汇报提纲

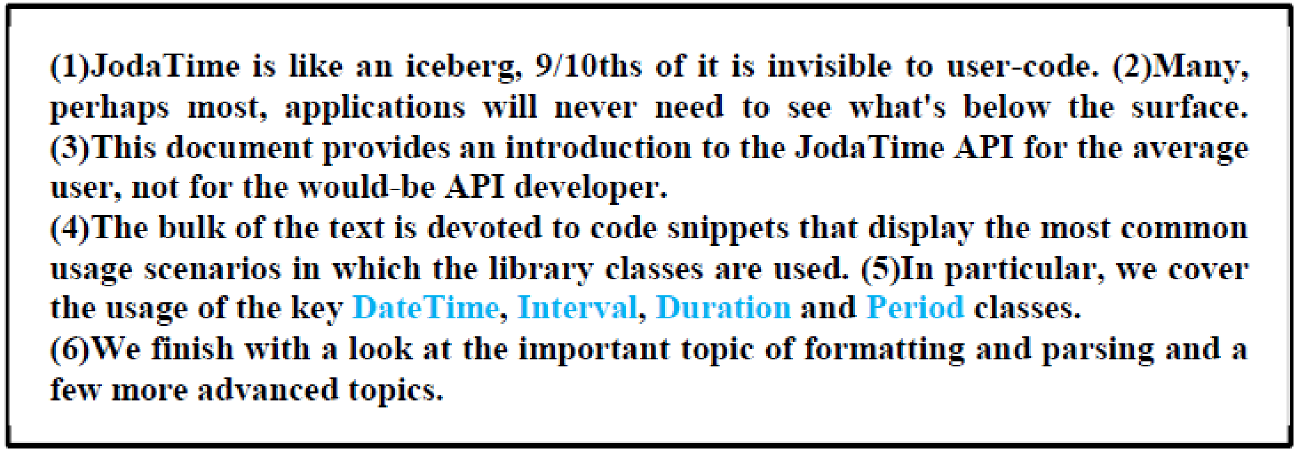
南京航空航天大学
Nanjing University of Aeronautics and Astronautics

API文档参考推荐

南京航空航天大学
Nanjing University of Aeronautics and Astronautics

问题：

- ✓ 单个API文档冗长，不能快速定位想要的内容。
- ✓ 解释某个API的文档片段分布在文档的各个部分。
- ✓ 并不是某个API出现，该文档就一定是解释这个API的。


(1)JodaTime is like an iceberg, 9/10ths of it is invisible to user-code. (2)Many, perhaps most, applications will never need to see what's below the surface. (3)This document provides an introduction to the JodaTime API for the average user, not for the would-be API developer.
(4)The bulk of the text is devoted to code snippets that display the most common usage scenarios in which the library classes are used. (5)In particular, we cover the usage of the key DateTime, Interval, Duration and Period classes.
(6)We finish with a look at the important topic of formatting and parsing and a few more advanced topics.

	IR	GMR	FITSEA
Input	Unfamiliar APIs		
Output	Relevant API tutorial fragments explaining unfamiliar APIs		
Method	Information Retrieval	Text Classification	Text Classification
Drawbacks	Precision is low	Different corpora require their corpus-specific annotated data. The effectiveness of supervised approaches depends on the features.	

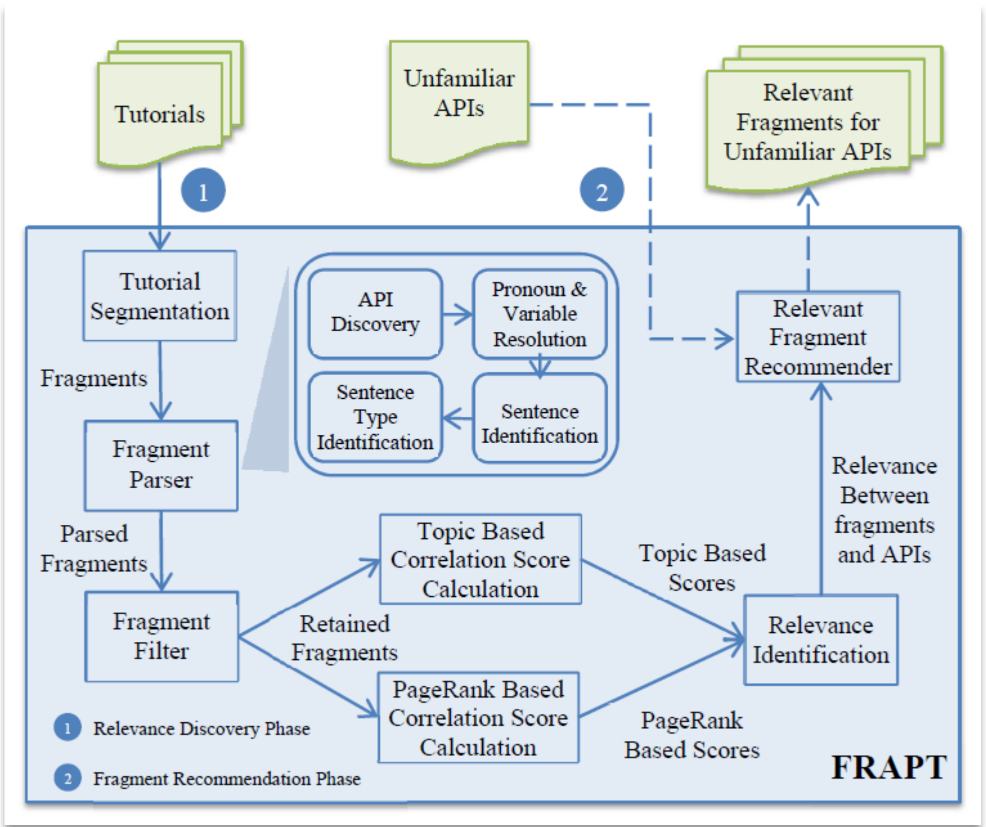
He Jiang, Jingxuan Zhang, Xiaochen Li, Zhilei Ren, and David Lo. A More Accurate Model for Finding Tutorial Segments Explaining APIs, *In Proc. of 23th IEEE International Conference on Software Analysis, Evolution, and Reengineering (SANER’ 2016)*, Osaka, Japan. pp. 157-167.

API文档参考推荐

南京航空航天大学 Nanjing University of Aeronautics and Astronautics

现有工作的不足：监督模型需要大量人工标注构建训练集，预测效果依赖于提取的特征，在应用中并不准确。

目标：提出一种无监督模型。



Tutorials
Unfamiliar APIs
Relevant Fragments for Unfamiliar APIs
1
2
Tutorial Segmentation
Fragments
Fragment Parser
Parsed Fragments
Fragment Filter
API Discovery
Pronoun & Variable Resolution
Sentence Type Identification
Sentence Identification
Relevant Fragment Recommender
Relevance Between fragments and APIs
Topic Based Correlation Score Calculation
Topic Based Scores
Retained Fragments
Relevance Identification
PageRank Based Correlation Score Calculation
PageRank Based Scores
1 Relevance Discovery Phase
2 Fragment Recommendation Phase
FRAPT

相关性识别阶段：发现API与文档片段的相关性。

片段推荐阶段：为用户输入的不熟悉的API推荐文档解释片段。

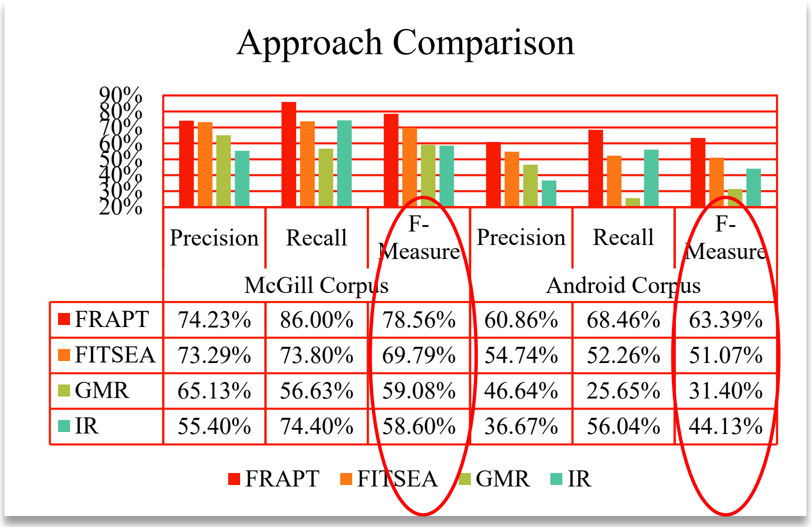
He Jiang, Jingxuan Zhang, Zhilei Ren, and Tao Zhang. An Unsupervised Approach for Discovering Relevant Tutorial Fragments for APIs, *In Proc. of 39th IEEE International Conference on Software Engineering (ICSE' 2017)*, Buenos Aires, Argentina. pp. 38-48.

API文档参考推荐

南京航空航天大学
Nanjing University of Aeronautics and Astronautics

我们提出的无监督方法效果如何?

Corpus	Tutorial	Precision (%)				Recall (%)				F-Measure (%)			
		FRAPT	FITSEA	GMR	IR	FRAPT	FITSEA	GMR	IR	FRAPT	FITSEA	GMR	IR
McGill Corpus	JodaTime	85.19	69.00	58.82	73.00	76.67	74.17	50.00	73.00	80.70	70.24	54.05	73.00
	Math Library	84.78	67.89	52.00	67.00	73.58	72.70	49.06	65.00	78.79	61.53	50.49	66.00
	Col. Official	62.03	55.74	62.69	30.00	87.50	48.62	31.79	94.00	72.59	48.10	42.18	45.00
	Col. Jenkov	61.19	90.44	82.14	33.00	97.62	85.17	58.97	88.00	75.23	85.17	68.66	48.00
	Smack	77.94	83.38	70.00	74.00	94.64	88.33	93.33	52.00	85.48	83.90	80.00	61.00
Android Corpus	Graphics	49.21	50.42	45.60	35.80	75.61	42.52	44.50	67.44	59.62	43.73	45.04	46.77
	Resources	65.22	75.83	55.00	40.32	66.67	66.17	21.11	55.56	65.93	66.80	30.51	46.73
	Data	71.43	56.52	19.29	33.33	55.56	52.00	14.76	44.00	62.50	54.17	16.72	37.93
	Text	57.58	36.19	66.67	37.21	76.00	48.33	22.22	57.14	65.52	39.56	33.33	45.07



	McGill Corpus			Android Corpus		
	Precision	Recall	F-Measure	Precision	Recall	F-Measure
🟥 FRAPT	74.23%	86.00%	78.56%	60.86%	68.46%	63.39%
🟧 FITSEA	73.29%	73.80%	69.79%	54.74%	52.26%	51.07%
🟩 GMR	65.13%	56.63%	59.08%	46.64%	25.65%	31.40%
🟦 IR	55.40%	74.40%	58.60%	36.67%	56.04%	44.13%

1. 作为一种无监督方法，我们提出的方法超过现有监督方法。
2. 考虑到无监督方法的优点，用我们提出的方法来发现解释API的文档片段是个很好的选择。

汇报提纲

南京航空航天大学
Nanjing University of Aeronautics and Astronautics

汇报总结

南京航空航天大学 Nanjing University of Aeronautics and Astronautics

- ✓ 作为学习API的最重要的资源，API文档的质量并不理想。
- ✓ 对API文档进行分析，进而提高API文档的质量是当前研究热点问题。
- ✓ 我们分别介绍了API文档内容解析，信息增强和参考推荐方面的工作。
- ✓ 但这些研究还无法完全消除API文档的全部问题，还有很多研究问题。

谢 谢 各 位 专 家

敬 请 批 评 指 正

张静宣

南京航空航天大学

2019年11月24日