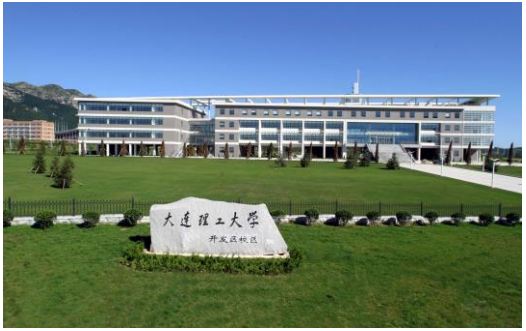
Semantic Estimation for Texts in Software Engineering

Reporter： Xiaochen Li
Dalian University of Technology, China

Oscar Lab

- Ph.D. candidate at OSCAR Lab, in Dalian University of Technology, China, under supervision with Prof. He Jiang from 2015. **OSCAR**: **O**ptimizing **S**oftware by **C**omputation from **AR**tificial intelligence



Oscar Lab

He Jiang

Lab Manager,
Ph.D., Professor

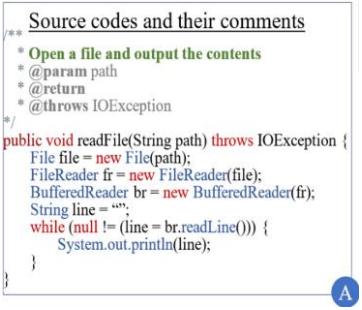
- 2 Professors
- 1 Associate Professor
- 1 Lecturer
- 7 PhD. Candidates
- 17 Master Students

- Mining software repositories
  - API mining
  - Crowd testing reports
  - Code search
  - Design pattern mining
  - Mobile APP mining
- Program & testing
  - Model checking
  - Complier optimization
- Search based software engineering
  - Next release problem
  - Software Task Allocation

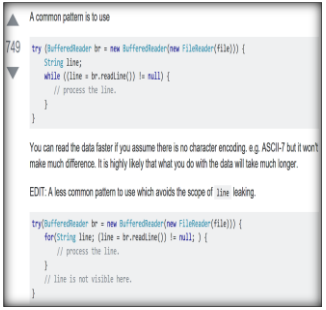
Overview

Texts in Software Engineering (SE)

Test logs



Codes & Comments



Q&A in forum

Bug reports

- > 4,200,000 test logs / year in industry
- > 300,000 projects in GitHub
- > 5,000,000 Q&A in Stack Overflow
- > 485,000 bug reports in Eclipse Repo.

Overview

Texts in Software Engineering (SE)

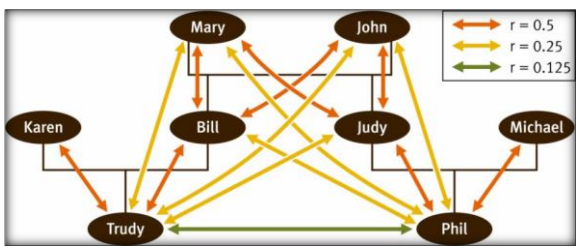
- Classify test logs
- Search APIs by queries
- Seek codes by asking questions
- Read bug reports
- ----

Texts mixed of Natural Language (**NL**) **words** and **APIs or codes** in Software Language (**SL**)

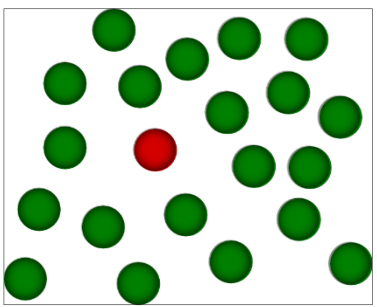
Overview

Semantic estimation for SE texts

- Given texts mixed natural language words and software APIs or codes,
  - how to estimate the similarity betw. texts?
  - how to find salient sentences in the text?



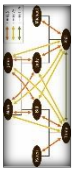
relatedness



importance

Overview

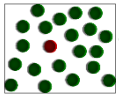
Semantic estimation work



relatedness

- **Cosine similarity+ KNN**
- Analyze the failure causes of test scripts

- **Word embedding**
- Recommend API sequences
- Link API documents to Ques.



importance

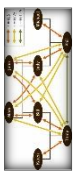
- **Crowdsourcing**
- Summarize bug reports

- **Deep neural network**
- Summarize bug reports

Shallow Bag-of-words → Deep Continuous spaces

Overview

Semantic estimation work



relatedness

- **Cosine similarity+ KNN**
- Analyze the failure causes of test scripts
- **Word embedding**
- Recommend API sequences
- Link API documents to Ques.

importance

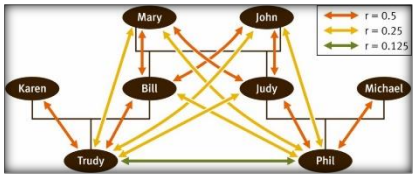
- **Crowdsourcing**
- Summarize bug reports
- **Deep neural network**
- Summarize bug reports

Shallow Bag-of-words → Deep Continuous spaces

Analyze failed test scripts

Semantic estimation work

- **Cosine similarity+ KNN**
- Analyze the failure causes of test scripts



1. **Why do we analyze failed test scripts?**
   - Failure causes are complex
   - Testers manually read logs to analyze
   - Logs are lengthy and complex
2. **How do we do that?**
   - Cosine similarity
   - KNN
3. **What are the results?**

Background

System and integration testing (SIT)

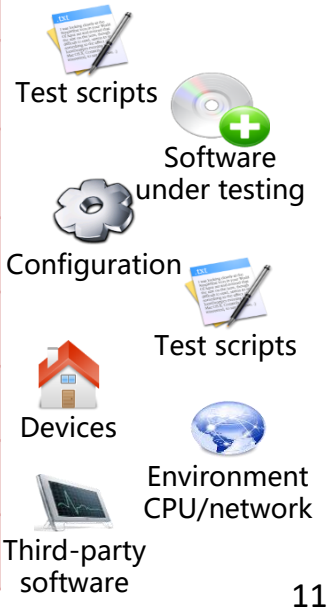
- Continuous integration increases SIT's frequency .
  - DevOps: faster time to market
  - Cloud-based system: run 1,000 test scripts in 25 minutes
- Running test scripts in SIT may fail.
  - We find 6000+ failures in a single month in one product
- Testers need to figure out the failure causes
  - Require the stakeholders to fix them

Background

Test alarms in SIT

- Test scripts may fail for various causes
  - A test alarm is an alarm to warn the test script failure

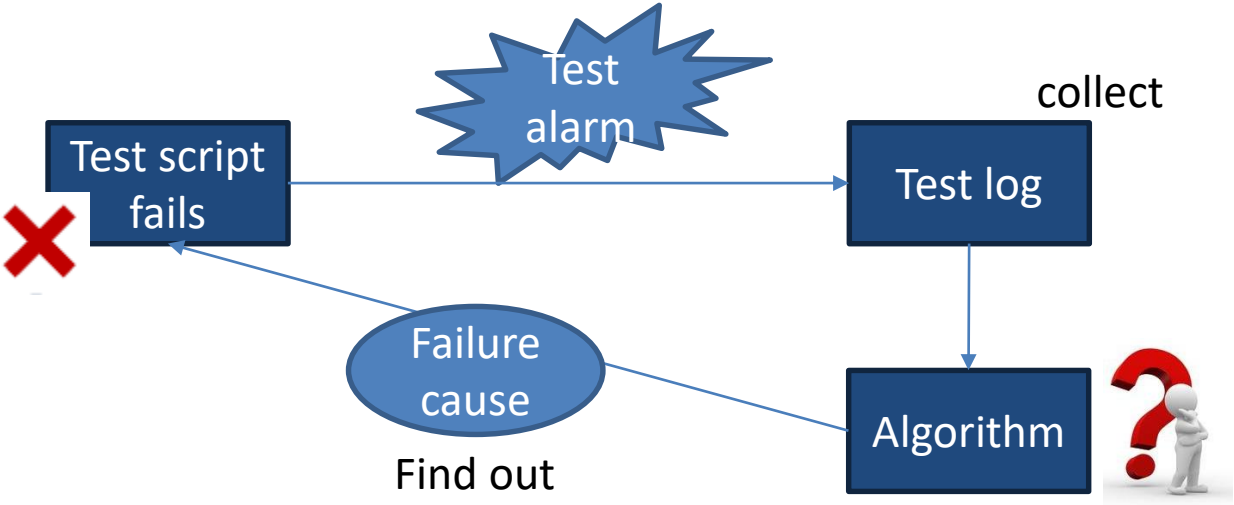
ID	Type of cause	Testers' solution
C1	Obsolete test	update test scripts
C2	Product code defect	submit bugs to developers
C3	Configuration error	correct configuration files
C4	Test script defect	debug test scripts
C5	Device anomaly	submit bugs to instrument suppliers
C6	Environment issue	diagnose the environment
C7	Software problem	ask site reliability engineers to diagnose



The Problem

Test alarm analysis

- Analyze the cause of test alarms (test script failure) by test logs
  - Test logs are easy to get
  - Testers also read test logs to analyze the alarms



The Problem

A test log

- Bilingual documents: English & Chinese
- Long: more than 1000 lines, more than 10GB (14,000 logs)

```
[2015-06-03 02:39:24.687]【189.106.7.11】[189.106.7.12 23]: cd /opt/VNFP/0

cd /opt/VNFP/0

-bash: cd: /opt/VNFP/0: No such file or directory
imageVMNPSO-001:~ #

[SSP_INFO] >>>>>>>>>>>>>>>>>>>>>rs cd /opt/VNFP/0
-bash: cd: /opt/VNFP/0: No such file or directory
imageVMNPSO-001:~ #  -#<D:/CIEnv_Fenix_HY/Test_Suit/IGP_DevM/SPC/common/Fenix/Feni

在测试步骤《查询虚拟机信息》中的检查点《获得vappid 成功》里出现断言失败
出错文件： C:/Program Files/Impeller/lib/ruby/lib/ruby/gems/1.8/gems/testlib-VrpBas
错误信息： 登录指定的文件夹失败
期望值：false
实际值：true
调用函数： assert_false
相关文件： D:/CIEnv_Fenix_HY/Test_Suit/IGP_DevM/SPC/common/Fenix/Fenix_mano.rb:373
```

1 Test step

2 Echo message

3

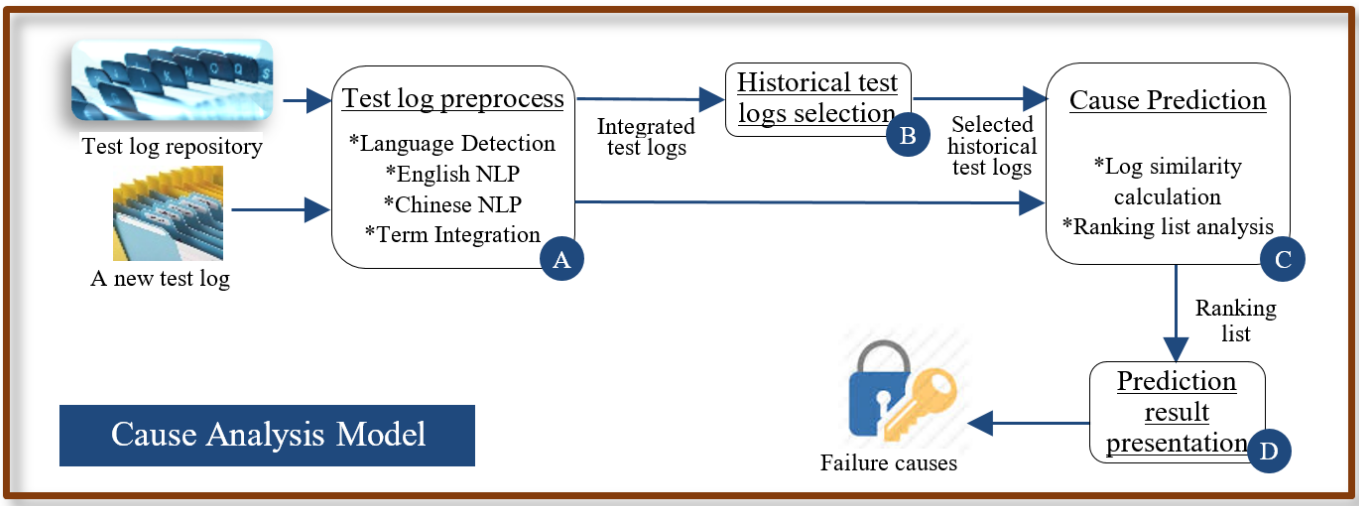
4 Miscellaneous message

5 Exception message

Cause Analysis Model (CAM)

Framework

- CAM's Idea
  - ➢ Search the test logs of historical test alarms that may have the same failure cause with the new test log



Cause Analysis Model (CAM)

Test log preprocess

- Language Detection

New test log snippet with function point "AUTO UPDATE SCHEMA (AUS)"
E [**exception happens continuously for more than 20 times**]
[2015-06-28 02:10:52.964] timed out while waiting for more data

Cause Analysis Model (CAM)

Test log preprocess

New test log snippet with function point "AUTO UPDATE SCHEMA (AUS)"
E [**exception happens continuously for more than 20 times**]
[2015-06-28 02:10:52.964] timed out while waiting for more data

- Language Detection
- English NLP
  - Tokenization,
  - Stop words removal

    (single letters, punctuation marks, and numbers),
  - Stemming

~~E [] [2015-06-28 02:10:52.964]~~ \ time~~d~~ \ ~~out~~ \ while \ wait~~ing~~ \ ~~for~~ \ more \ data

Cause Analysis Model (CAM)

Test log preprocess

New test log snippet with function point "AUTO UPDATE SCHEMA (AUS)"
E [**exception happens continuously for more than 20 times**]
[2015-06-28 02:10:52.964] timed out while waiting for more data

- Language Detection
- English NLP
  - Tokenization,
  - Stop words removal

    (single letters, punctuation marks, and numbers),
  - Stemming
- Chinese NLP
  - Word segmentation

~~E [] [2015-06-28 02:10:52.964]~~ \ time~~d~~ \ ~~out~~ \ while \ wait~~ing~~ \ ~~for~~ \ more \ data

exception \ happens \ continuously \ for more than \ ~~20~~ \times

Cause Analysis Model (CAM)

Test log preprocess

New test log snippet with function point "AUTO UPDATE SCHEMA (AUS)"
E [**exception happens continuously for more than 20 times**]
[2015-06-28 02:10:52.964] timed out while waiting for more data

- Language Detection
- English NLP
  - Tokenization,
  - Stop words removal

  (single letters, punctuation marks, and numbers),
  - Stemming

~~E [] [2015-06-28 02:10:52.964]~~ \ time~~d~~ \ ~~out~~ \ while \ wait~~ing~~ \ ~~for~~ \ more \ data

- Chinese NLP
  - Word segmentation

exception \ happens \ continuously \ for more than \ ~~20~~ \times

- Term Integration

bag-of-words

**exception \ happens \ continuously \ for more than \ times ** time \ while \ wait \ more \ data

Cause Analysis Model (CAM)

Cause prediction

- Log similarity with historical logs
  - 2-shingling terms (successfully applied in information retrieval)
  - TF-IDF based cosine similarity

**exception \ happens \ continuously \ for more than \ times **
time \ while \ wait \ more \ data

**exception happens **
**happens continuously **
**continuously for more than **
**for more than times **
times time \
time while \
while wait \
wait more \
more data

Logs	Func. Point	Sim_{log}	Cause
his3	AUS	0.586	C2
his4	AUS	0.472	C3
his1	AUS	0.322	C3
his2	AUS	0.320	C3
his5	AUS	0.134	C2

Cause Analysis Model (CAM)

Cause prediction

- Predict by k-Nearest Neighbor
  - Case 1: the similarity of top 1 log (his3) exceeds a threshold
  - Case 2: the similarity of top 1 log (his3) is lower than a threshold
    - C2=0.586+0.134; **C3**=0.472+0.311+0.320

Case 1 threshold=0.5

Logs	Func. Point	Sim_{log}	Cause
his3	AUS	0.586	**C2**
his4	AUS	0.472	C3
his1	AUS	0.322	C3
his2	AUS	0.320	C3
his5	AUS	0.134	C2

Case 2 threshold=0.6

Logs	Func. Point	Sim_{log}	Cause
his3	AUS	0.586	C2
his4	AUS	0.472	C3
his1	AUS	0.322	C3
his2	AUS	0.320	C3
his5	AUS	0.134	C2

Experimental Setup

- Two industrial testing projects at Huawei-Tech Inc.
  - **14,000 test logs** of failed test scripts, **manually labeled**

- **Evaluation method**
  - Accuracy、Area-Under-Curve
  - Running time, memory consumption
  - Incremental framework (simulate testers' daily work)

- Baseline Algorithms: **bag-of-words**
  - Lazy Associative Classifier (LAC)
  - Best First Tree (BFT).
  - Topic Model (TM)

Experimental Results

Overall performance

- How does CAM perform against baseline algorithms?

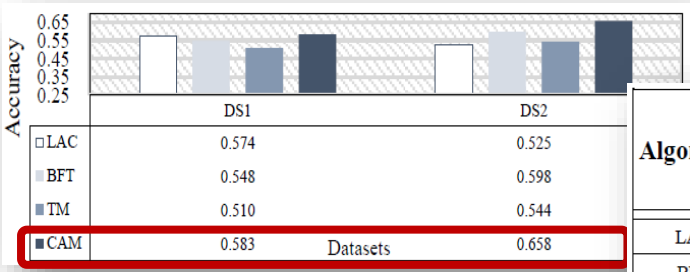


Fig. 1 Accuracy

Algorithm	Time (in minutes)						Memory	
	DS1 (7356 test logs)			DS2 (6557 test logs)			DS1	DS2
	Training	Test	Total	Training	Test	Total		
LAC	11.4	1	12.4	3.6	1.4	5	3 GB	3 GB
BFT	208.6	0.3	208.9	46.8	0.2	47	22 GB	20 GB
TM	75.1	2.8	77.9	142	4.3	146.3	8 GB	5 GB
CAM	0	6.9	6.9	0	14.4	14.4	4 GB	4 GB

Fig. 2 Comparison on computation resources

- Outperform the baseline algorithms ($p<0.05$)
- Superior over the majority of cause types
- Resources saving, take about 0.1s and less than 4GB memory to process a test log.

Experimental Results

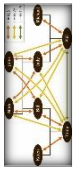
Evaluation in real scenario

- How does CAM perform in a real development scenario?
  - 72% accuracy after running for two months.

- Feedback
  - CAM is better than manually building regular expressions.
  - Actually, I will not believe in an automatic tool. However, after presenting the historical test logs, I can quickly decide whether the prediction is correct. **CAM accelerates my work**.
  - **Suggestions**: labeling the defect-related snippets, provide suggestions on how to fix defects

Overview

Semantic estimation work



relatedness

- **Cosine similarity+ KNN**
- Analyze the failure causes of test scripts
- **Word embedding**
- Recommend API sequences
- Link API documents to Ques.

importance

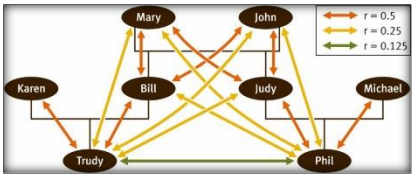
- **Crowdsourcing**
- Summarize bug reports
- **Deep neural network**
- Summarize bug reports

Shallow Bag-of-words → Deep Continuous spaces

Word2API

Semantic estimation work

- **Word embedding**
- Recommend API sequences
- Link API documents to Ques.



1. **Why do we need word embedding?**
   - Relatedness between words and APIs
   - Better than bag-of-words
2. **How do we do that?**
   - Collect large documents having words & APIs
   - Word embedding
3. **What are the results?**

Background

Semantic gaps

- Gaps between natural languages and APIs
  - High-level vs. Low-level
  - For example: *read a file*

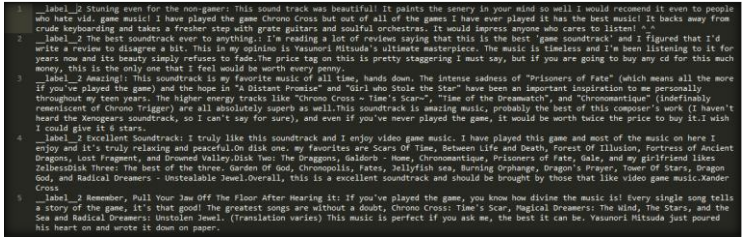
```
File file = new File();
FileReader fr = new FileReader(file);
BufferedReader br = new BufferedReader(fr);
String line = "";
while (null != (line = br.readLine()) {
    System.out.println(line);
}
```

- java.io.File#new,
- java.io.FileReader#new,
- java.io.BufferedReader#new,
- java.io.BufferedReader#readLine

Word Embedding

Words into low-dimension vectors

- Easy to implement
  - Prepare a dataset



  - Word2Vec Tool

  - Run CBOW or Skip-gram

$Word_1$:	<0.365,0.846,0.025…>
$Word_2$:	<0.585,0.298,0.525…>
API_1:	<0.316,0.003,0.298…>
API_2:	<0.337,0.723,0.221…>
…	…

Word Embedding

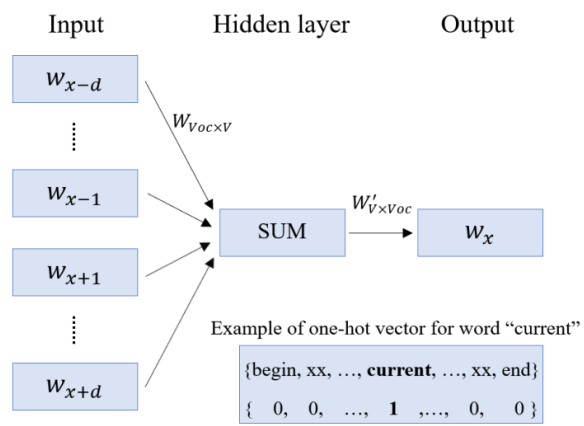
Continuous Bag-of-Words model CBOW

- Minimize differences between output and w_x

Interface IPageLayout
Description: A page layout defines the initial layout perspective within a page in a workbench window... View placeholders may also have a secondary id. ... For example, the placeholder "someView:*" will match any occurrence of the view that has primary id "someView" and that also has some non-null secondary id. Note that this placel not match the view if it has no secondary id ...

word2vec

$$L_M = \frac{1}{X}\sum_{x=1}^{X} \log p(w_x | W_x^d)$$



Word Embedding

Challenge

- Acquisition challenge
  - how to collect large numbers of documents that contain diversity words and APIs

> Interface `IPageLayout`
> Description: A page layout defines the initial layout for a perspective within a page in a workbench window... View placeholders may also have a secondary id. ... For example, the placeholder "someView:*" will match any occurrence of the view that has primary id "someView" and that also has some non-null secondary id. Note that this placeholder will not match the view if it has no secondary id ...

  - org.w3c.dom.views.DocumentView#getDefaultView()
  - java.x.swing.text.View.ComponentView#new()

Word Embedding

Challenge

- Alignment challenge
  - how to make semantically related words and APIs co-occur in a fixed window size

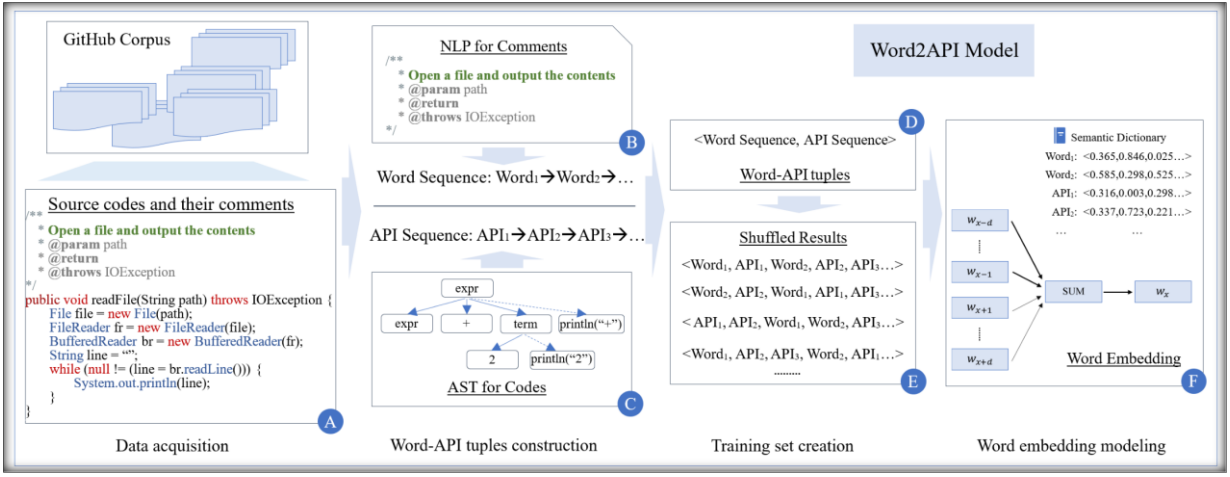
word2vec

Interface `IPageLayout`
Description: A page layout defines the initial layout for a perspective within a page in a workbench window... View placeholders may also have a secondary id. ... For example, the placeholder "someView:*" will match any occurrence of the view that has primary id "someView" and that also has some non-null secondary id. Note that this placeholder will not match the view if it has no secondary id ...

Word Embedding

Word2API

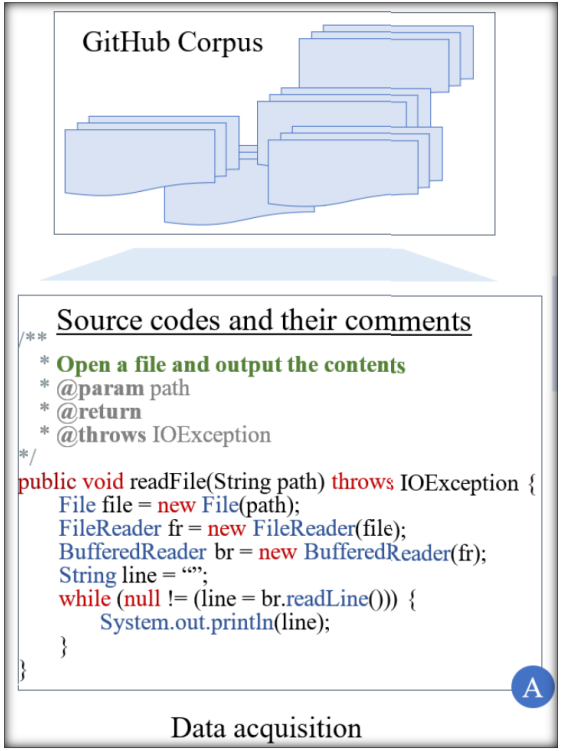
- Collect source codes and APIs from GitHub (**acquisition**)
- Pre-process words & APIs with NLP and Abstract Syntax Trees
- Shuffle words and APIs (**alignment**)
- Run Word Embedding Modeling



Word2API

Data acquisition

- GitHub from 2008-2016
  - 391,690 Java projects
  - 31,211,030 source code files
  - Many words and APIs that developers used



Word2API

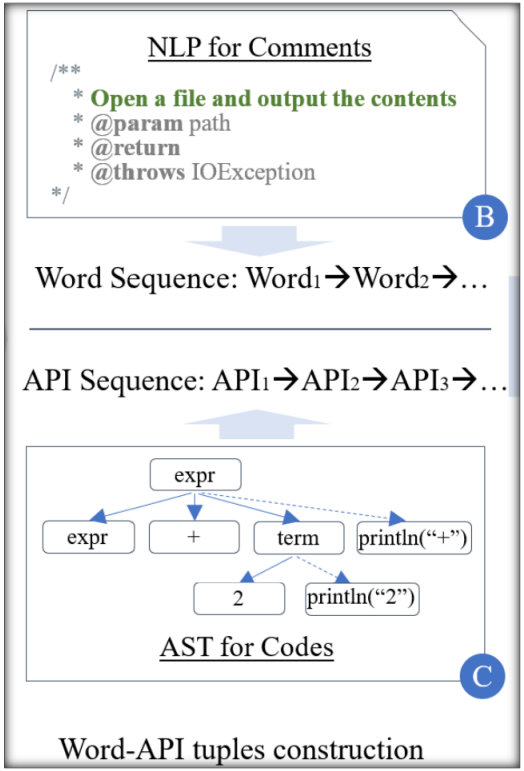
Word-API Tuples Construction

- NLP
  - tokenization,
  - Stop word removal,
  - Stemming

open a file and output the contents

Word sequence

<open, file, output, content>



Word-API tuples construction

Word2API

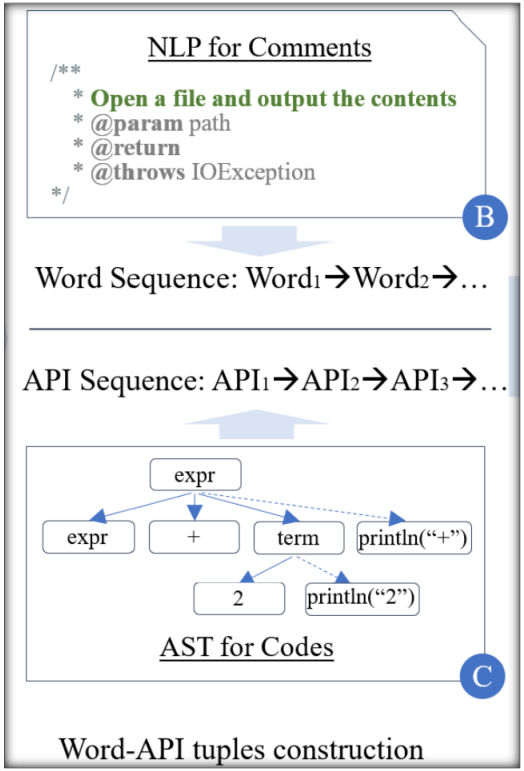
Word-API Tuples Construction

- AST (Abstract Syntax Trees)
- Finding API fully qualified name in the text

```
public void readFile(String path) throws IOException {
    File file = new File(path);
    FileReader fr = new FileReader(file);
    BufferedReader br = new BufferedReader(fr);
    String line = "";
    while (null != (line = br.readLine())) {
        System.out.println(line);
    }
}
```

API Sequence

<java.io.File#new,
java.io.FileReader#new,
java.io.BufferedReader,
java.lang.String#new,
java.io.BufferedReader#readLine,
....>

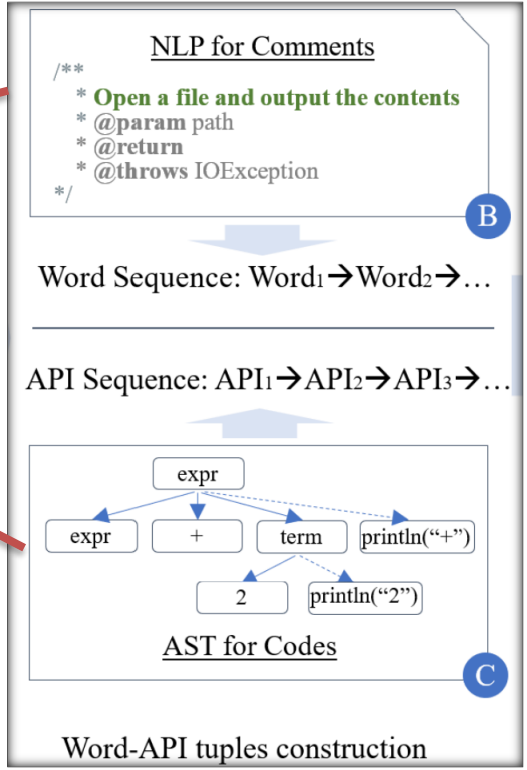


Word2API

Word-API Tuples Construction

- 13,883,230 tuples

<word1, word2, …, API1, API2…>



Word2API

Training Set Creation

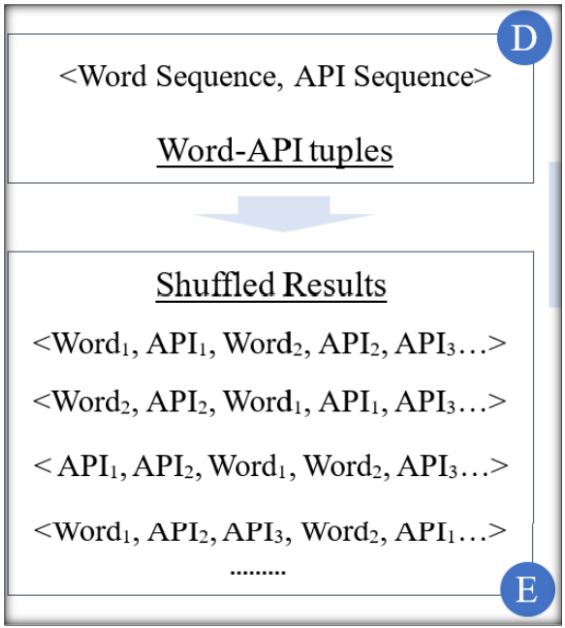
- 13,883,230 tuples

<word1, word2, word3, word4, word5,…, API1, API2, API3…>

Word2API

Training Set Creation

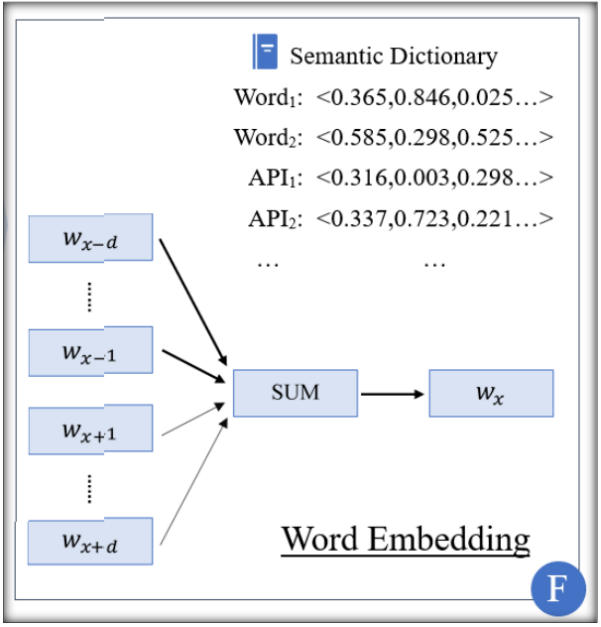
- The underlying reason of the above procedure is that if a word and an API are semantically related, they tend **to co-occur in the same tuple**. After shuffling, the related words and APIs will have **higher chances to locate in the same window** than unrelated ones when the corpus is a large
- 138,832,300 shuffled results
- >30 GigaByte.



Word2API

Word Embedding Modeling

- 87,270 word vectors
- 37,431 API vectors
- Semantic estimation with these vectors



- Word-API similarity

$$sim(w,a) = \frac{\overrightarrow{V_w} \cdot \overrightarrow{V_a}}{|\overrightarrow{V_w}||\overrightarrow{V_a}|}.$$

- Words-APIs similarity

$$sim(W,A) = \frac{1}{2}\left(\frac{\sum(maxSim(w,A) \times idf(w))}{\sum idf(w)} + \frac{\sum(maxSim(a,W) \times idf(a))}{\sum idf(a)}\right),$$

Application 1

Query augmentation

- For API Sequences Recommendation

Query

"read a file"

API Seq.

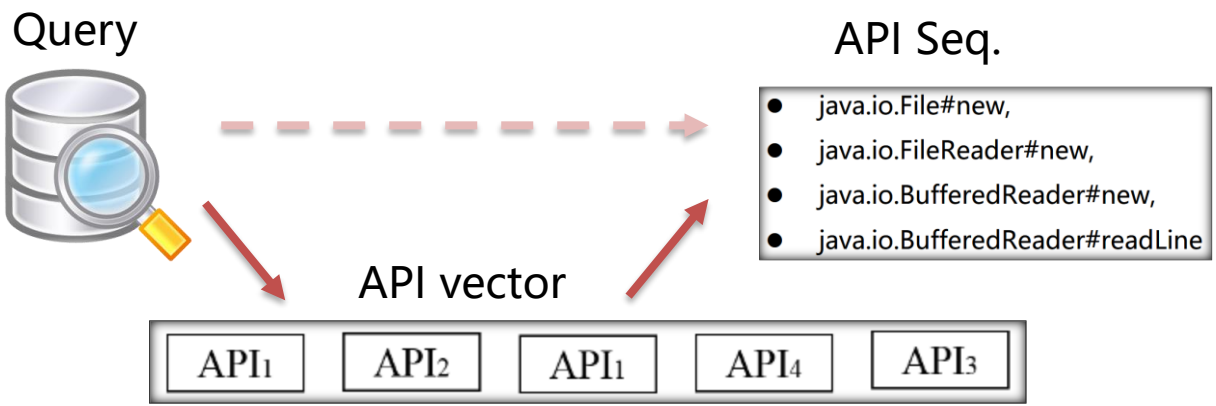
- java.io.File#new,
- java.io.FileReader#new,
- java.io.BufferedReader#new,
- java.io.BufferedReader#readLine

These sequences are retrieved from source code corpus, e.g. GitHub corpus.

Application 1

Query augmentation algorithms

- Augment queries into API vectors



- SWIM: Word Alignment based Augmentation
- CodeHow: API Description based Augmentation
- Word2API based Augmentation

Application 1

TABLE III: Performance of query augmentation algorithms over 30 human written queries.

ID	Query	SWIM			CodeHow			Word2API		
		FR	RR5	RR10	FR	RR5	RR10	FR	RR5	RR10
Q1	convert int to string	11	0	0	11	0	0	3	0.2	0.1
Q2	convert string to int	1	1	0.5	11	0	0	1	0.8	0.8
Q3	append string	1	1	1	1	1	1	1	1	1
Q4	get current time	11	0	0	11	0	0	1	1	1
Q5	parse datetime from string	10	0	0.1	11	0	0	1	1	0.7
Q6	test file exists	1	1	1	1	1	1	1	0.8	0.8
Q7	open a url	1	1	1	1	1	1	1	0.8	0.8
Q8	open file dialog	11	0	0	1	0.8	0.7	1	0.4	0.7
Q9	get files in folder	11	0	0	1	0.8	0.9	1	1	0.9
Q10	match regular expressions	1	1	0.8	1	0.6	0.7	1	1	1
Q11	generate md5 hash code	11	0	0	11	0	0	1	1	1
Q12	generate random number	1	0.4	0.2	1	1	1	1	1	1
Q13	round a decimal value	11	0	0	2	0.2	0.1	1	0.8	0.8
Q14	ex[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	2	0.6	0.5
Q15	co[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	1	1	1
Q16	cre[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	1	1	1
Q17	co[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	1	0.6	0.5
Q18	copy a file and save it to your destination path	1	1	1	2	0.2	0.3	1	0.8	0.9
Q19	delete files and folders in a directory	1	1	1	3	0.6	0.4	4	0.4	0.4
Q20	reverse a string	11	0	0	11	0	0	11	0	0
Q21	create socket	11	0	0	1	0.6	0.4	1	1	0.9
Q22	rename a file	11	0	0	11	0	[illegible]	4	0.4	0.5
Q23	download file from url	1	1	0.7	1	1	1	5	0.2	0.3
Q24	serialize an object	1	1	1	1	1	1	1	1	1
Q25	read binary file	1	1	0.6	1	1	1	1	0.8	0.8
Q26	save an image to a file	1	1	1	1	1	1	5	0.2	0.4
Q27	write an image to a file	1	1	1	1	0.8	0.6	2	0.4	0.3
Q28	parse xml	11	0	0	11	0	0	1	0.2	0.3
Q29	play audio	11	0	0	1	0.8	0.9	1	0.4	0.5
Q30	play the audio clip at the specified absolute URL	11	0	0	1	1	1	1	0.6	0.4
	Average scores over 30 queries	5.633	0.513	0.463	4.467	0.547	0.533	1.933	0.680	0.677

Position of first correct API seq. : lower is better

Ratio of correct API seq.: higher is better

Application 2

API documents linking

- Link API documents with Stack Overflow questions
  - Question: "**Are there any good CachedRowSet implementations other than the proprietary Sun one?**"

Linkage

13

You shouldn't be directly instantiating implementation of CachedRowSet -- use its Provider to obtain an instance: see http://docs.oracle.com/javase/7/docs/api/javax/sql/rowset/RowSetProvider.html (available since JDK7)

In fact, CachedRowSet's interface and related

Overview Package Class Use Tree Deprecated Index Help

Prev Class Next Class Frames No Frames All Classes

Summary: Nested | Field | Constr | Method Detail: Field | Constr | Method

javax.sql.rowset

Class RowSetProvider

java.lang.Object
javax.sql.rowset.RowSetProvider

```
public class RowSetProvider
extends Object
```

A factory API that enables applications to obtain a RowSetFactory implementation that can be used to create different types of RowSet implementations.

Example:

```
RowSetFactory aFactory = RowSetProvider.newFactory();
CachedRowSet crs = aFactory.createCachedRowSet();
...
RowSetFactory rsf = RowSetProvider.newFactory("com.sun.rowset.RowSetFactoryImpl", null);
WebRowSet wrs = rsf.createWebRowSet();
```

Tracing of this class may be enabled by setting the System property javax.sql.rowset.RowSetFactory.debug to any value but false.

Since:

1.7

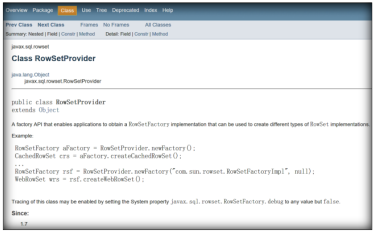
Application 2

Word2API for API Doc. Linking

- Collect words in the question
  - *are there any good CachedRowSet implementations other than the proprietary Sun one*

$$sim(W, A) = \frac{1}{2}\left(\frac{\sum (maxSim(w, A) \times idf(w))}{\sum idf(w)} + \frac{\sum (maxSim(a, W) \times idf(a))}{\sum idf(a)}\right),$$

- Collect APIs in API documents
  - *javax.sql.rowset.RowSetProvider#newFactory*
  - *javax.sql.rowset.RowSetProvider#createCachedRowSet*
  - *……*



Application 2

Results

- MAP: Mean Average Precision
- MRR: Mean Reciprocal Rank

TABLE V: MAP and MRR for API document linking.

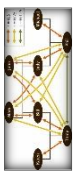
Algorithms	MAP	MRR
VSM	0.232	0.259
Embedding	0.313	0.354
Word2API	0.402	0.433
VSM+Embedding	0.340	0.380
VSM+Word2API	0.436	0.469

- Algorithms
  - VSM: bag-of-words
  - Embedding: previous work
  - VSM+XXX: combined

1. Word2API can bridge gaps betw. NL and SL
2. Word Embedding is better that bag-of words here
3. We can combine different techniques for better results

Conclusion

Semantic estimation work



relatedness

- **Cosine similarity+ KNN**
- Analyze the failure causes of test scripts
- **Word embedding**
- Recommend API sequences
- Link API documents to Ques.

importance

- **Crowdsourcing**
- Summarize bug reports
- **Deep neural network**
- Summarize bug reports

Shallow Bag-of-words → Deep Continuous spaces

Thanks

Reporter： Xiaochen Li
Dalian University of Technology, China