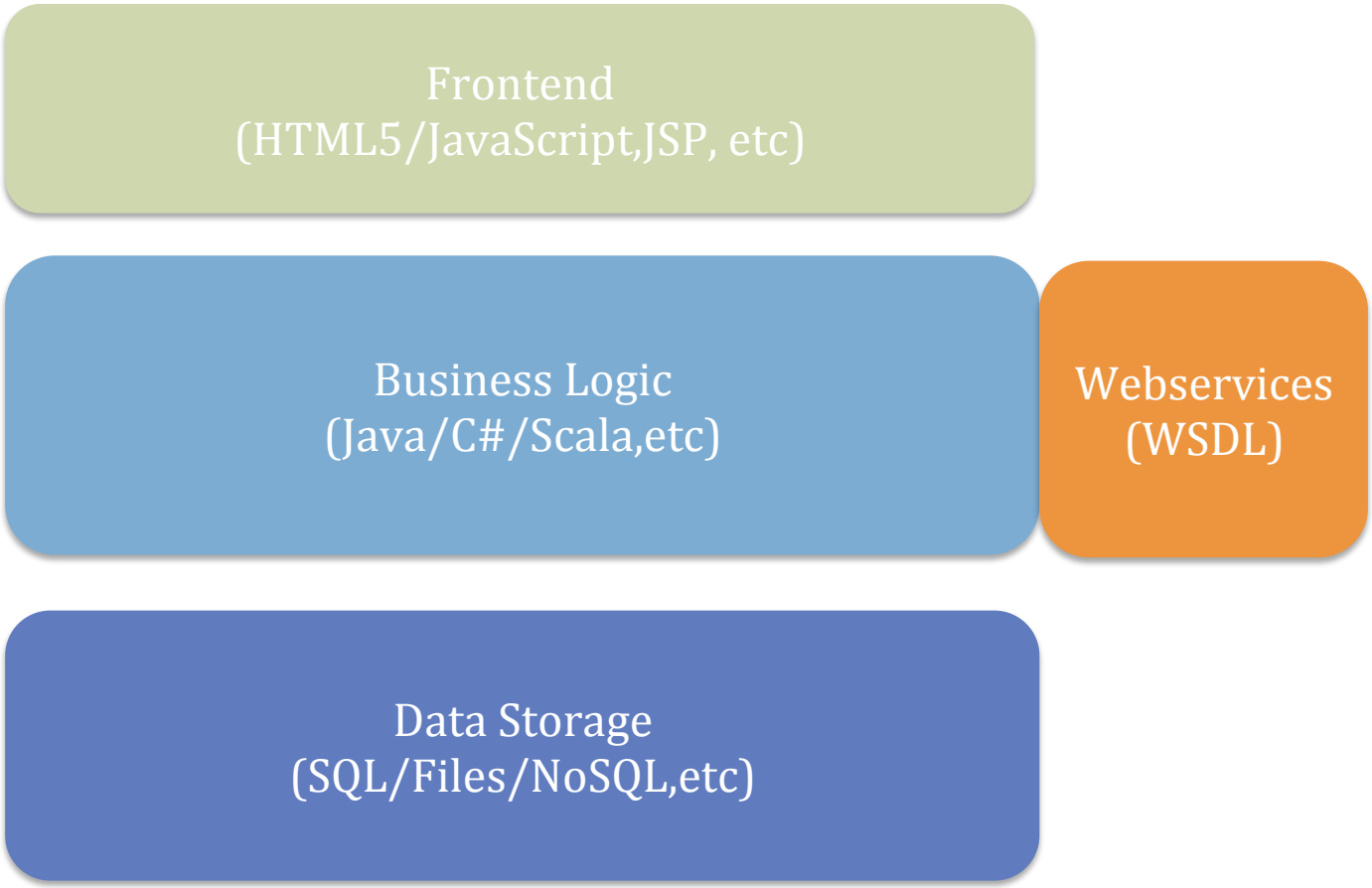
# Detecting Cross-language Dependencies Generically

Theodoros Polychniatis, Eric Bouwers,
Joost Visser, Jurriaan Hage, Slinger Jansen

Universiteit Utrecht

# The Problem


Frontend
(HTML5/JavaScript,JSP, etc)
Business Logic
(Java/C#/Scala,etc)
Webservices
(WSDL)
Data Storage
(SQL/Files/NoSQL,etc)

# Some conditions

Accuracy

Explainability

Extensibility

Static analyses

Incomplete systems

http://www.freeiconsweb.com

# The basic idea

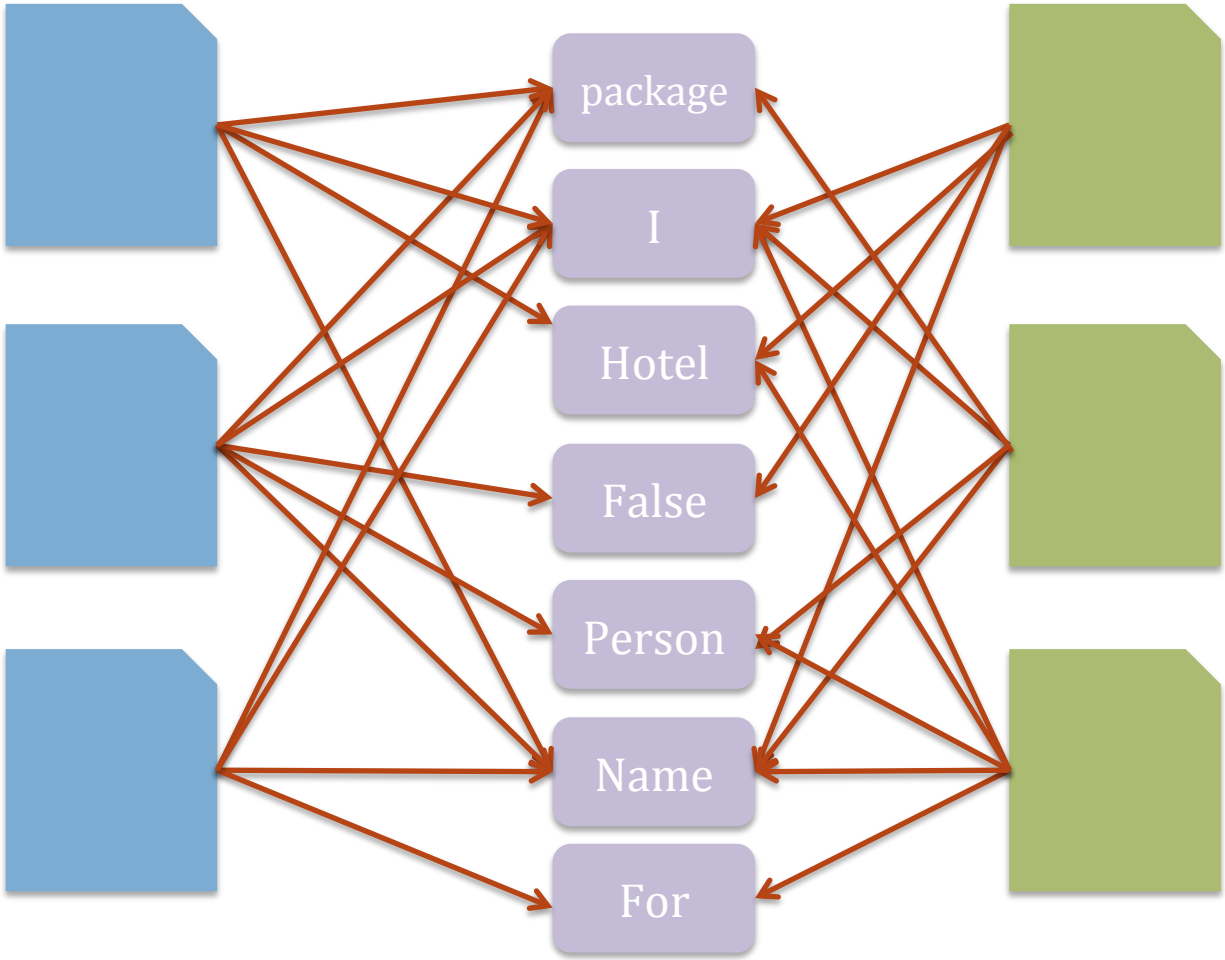
```
1:  create a graph of the system's source modules
2:  for each module in the graph do
3:      replace comments and special characters in the module's
        content with spaces
4:      tokenize the text by splitting it on white space
5:      extract all the tokens (words)
6:      for each token do
7:          check the graph for a node with the name of the token
8:          if token-node does not exist then
9:              create a node with the same name as the token
10:         end if
11:         create an edge from the module to the token-node
12:     end for
13: end for
14: for each token-node do
15:     if node is connected with fewer than two modules from
        different languages then
16:         remove node
17:     end if
18: end for
```

Get all tokens from source-modules

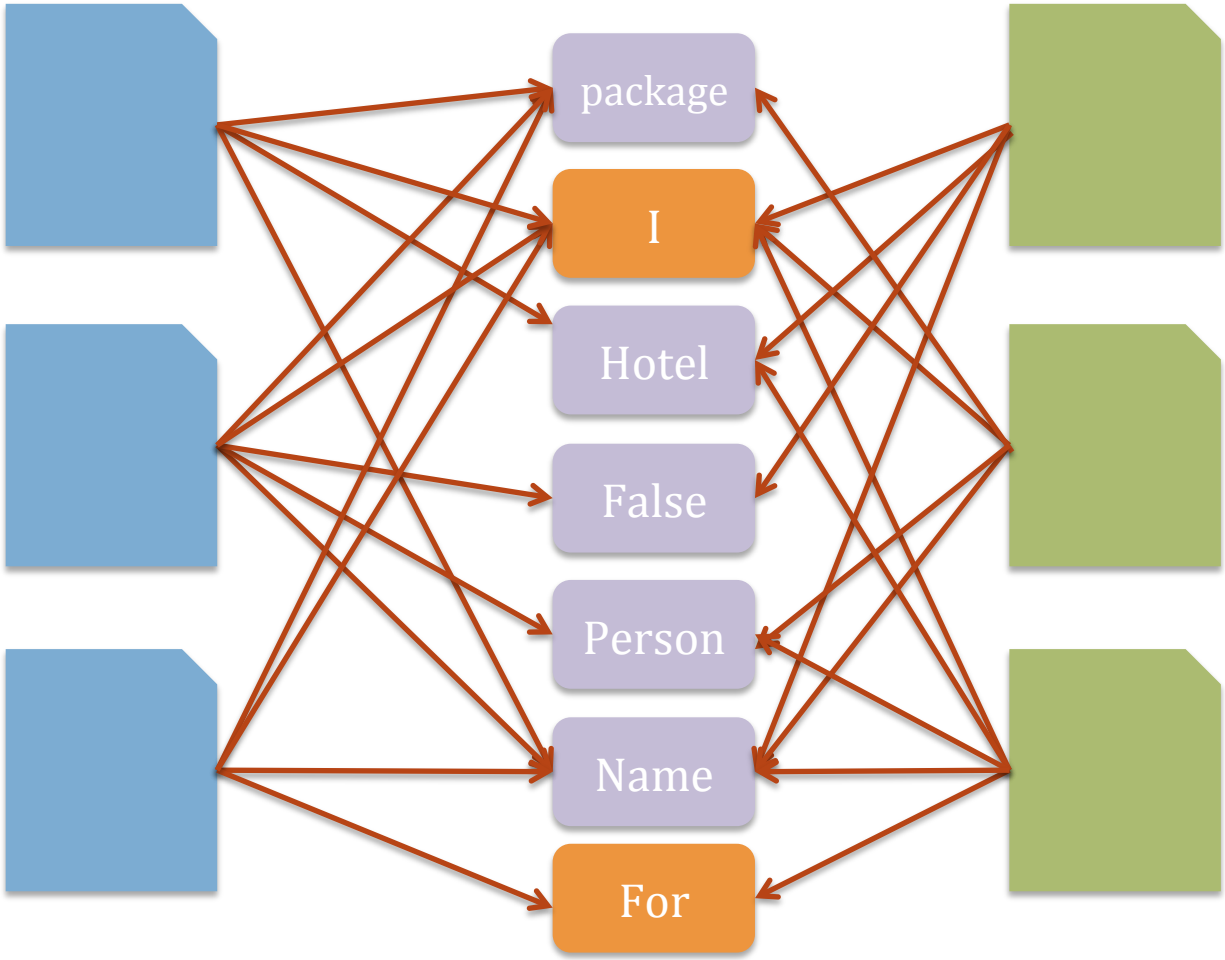
Link source-modules based on tokens

Remove all tokens that do not appear across languages

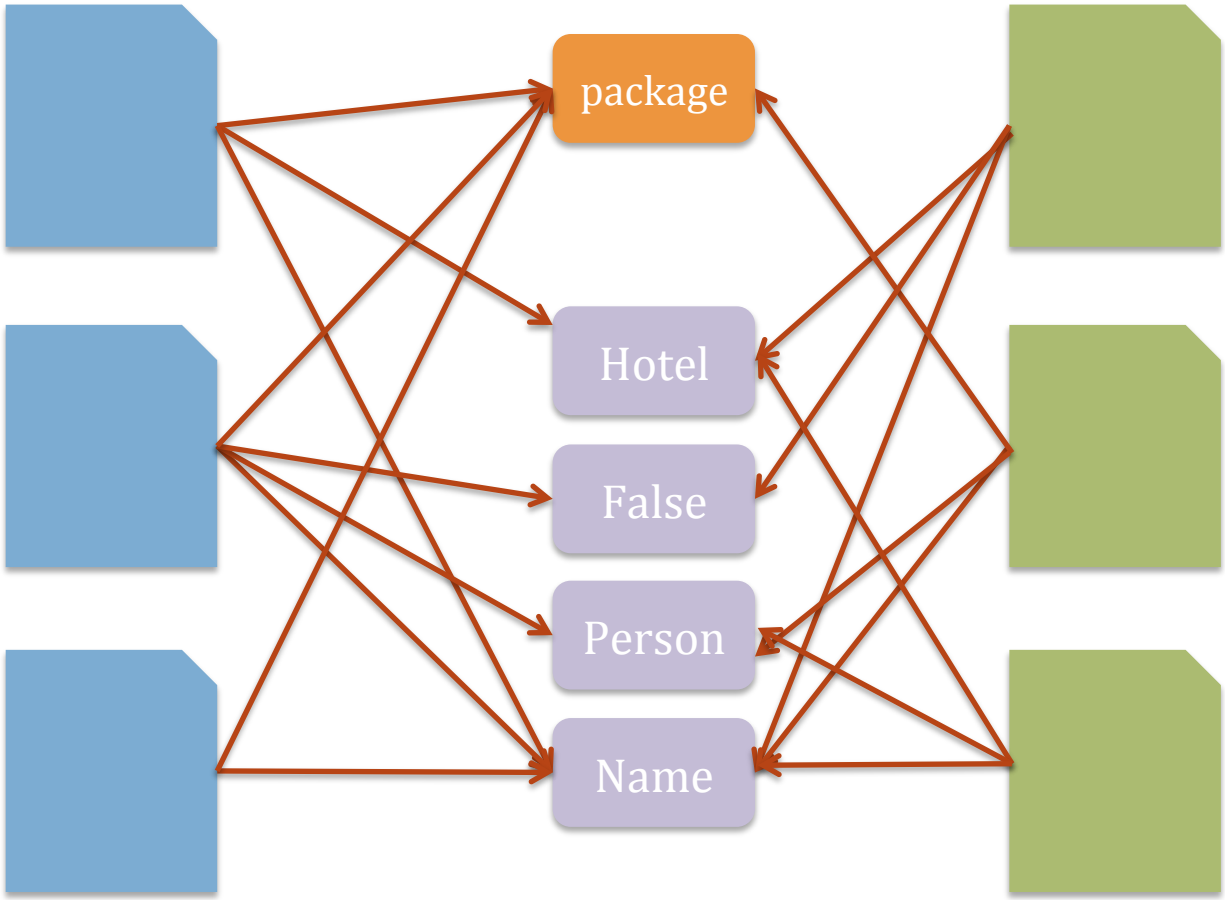
# The Initial Result



# Step 1 – filter out frequent tokens

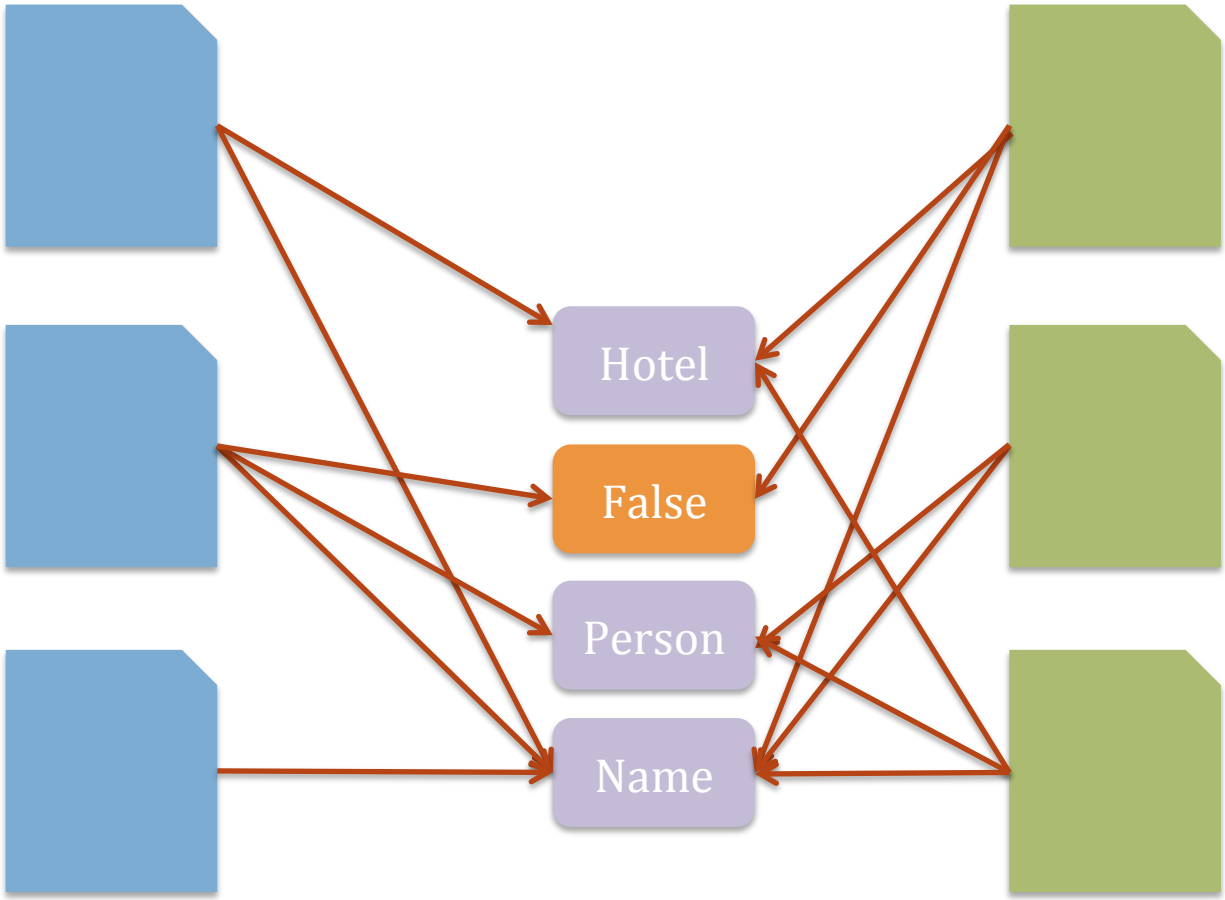


# Step 2 – language frequent tokens

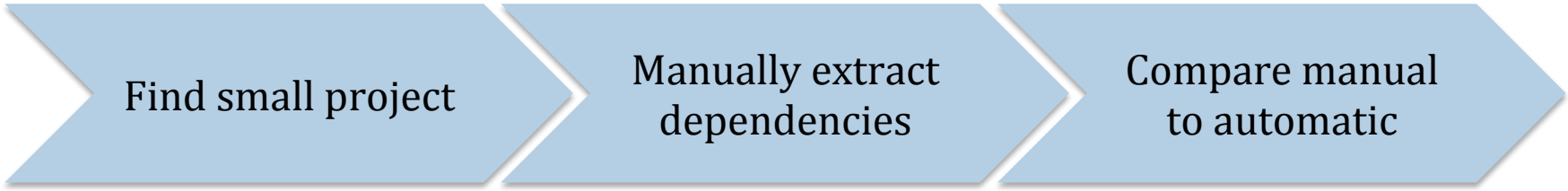



$$\frac{\#modules\ in\ a\ language\ that\ contain\ the\ token}{Total\ \#modules\ in\ this\ language}$$

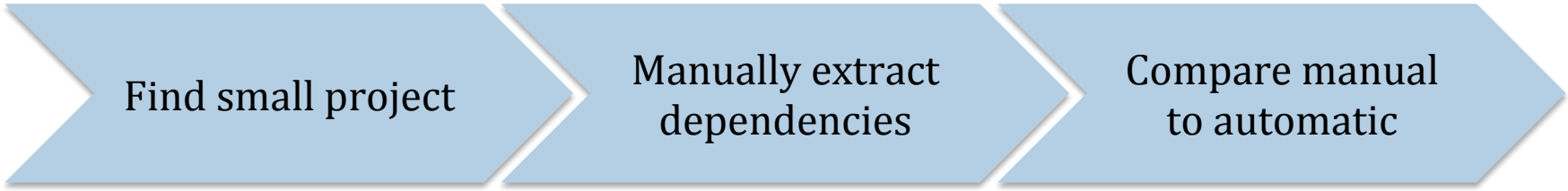
# Step 3 – filter low weight tokens



# But is this feasible?

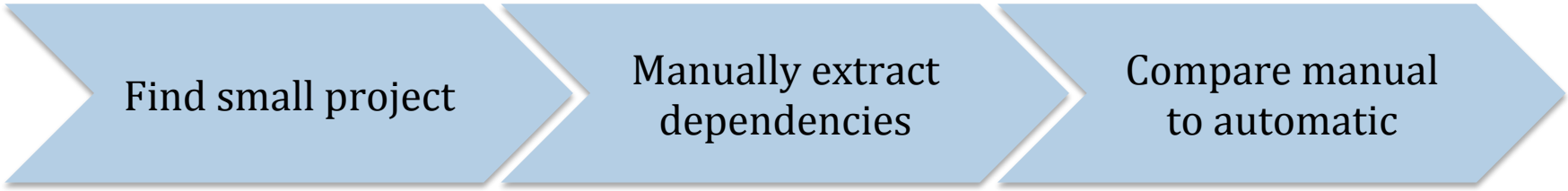



# But is this feasible?



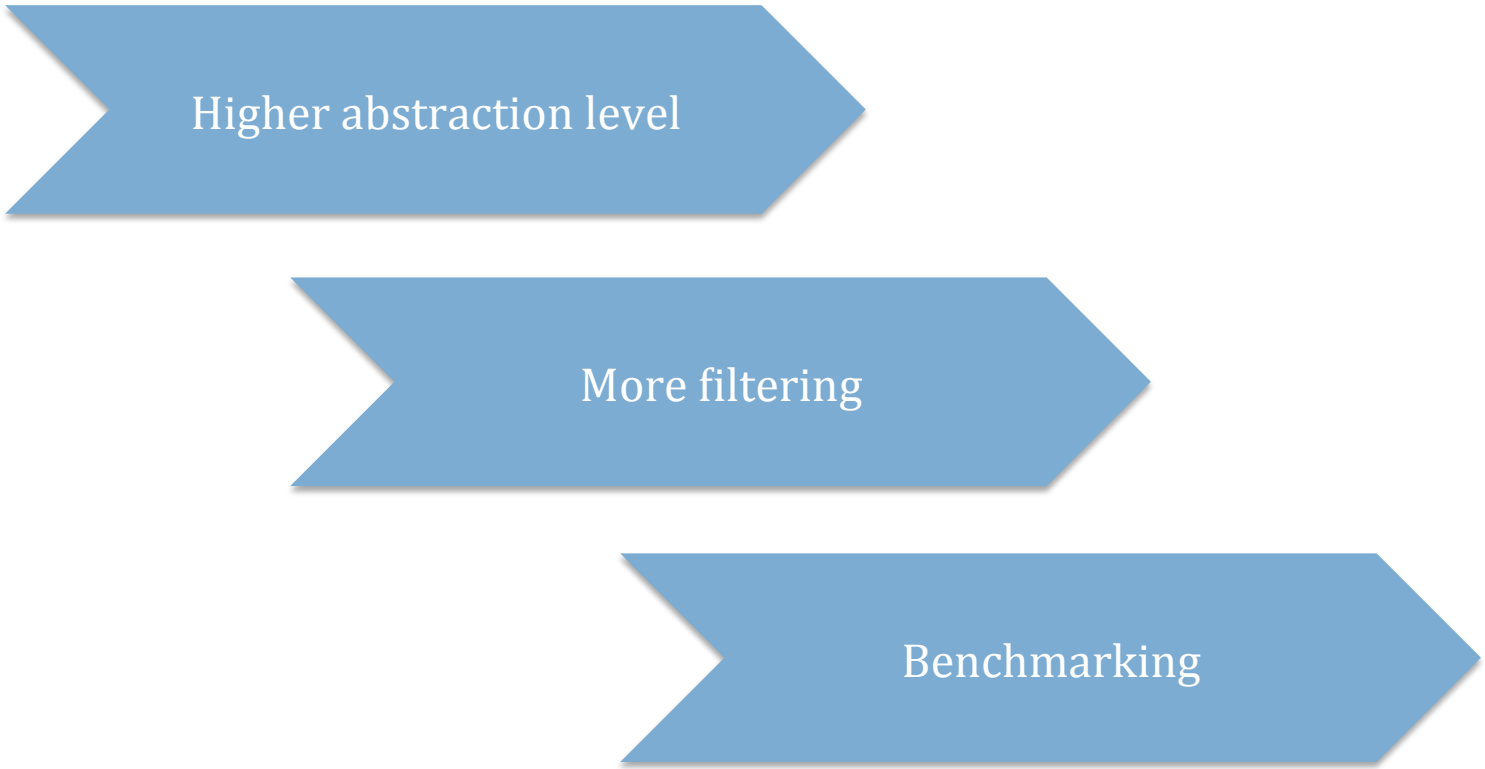

| **Attributes** | **Value** |
|---|---|
| Direct dependencies | 6 |
| Framework dependencies | 53 |
| Indirect dependencies | 35 |
| Total relevant dependencies | 94 |
| Non-dependent file combinations | 306 |
| Collection (*All possible file combinations*) | 400 |

# But is this feasible?

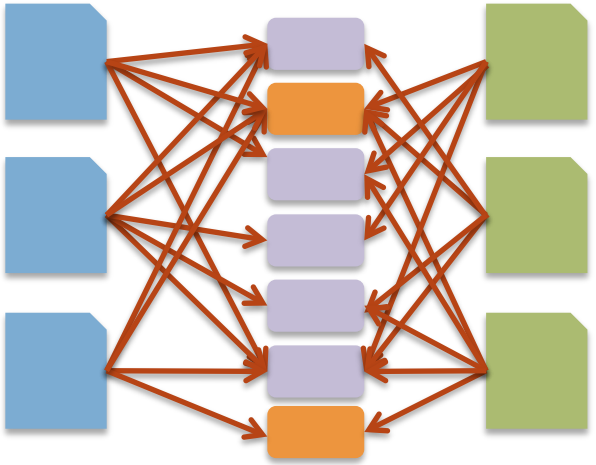



| Recall | Precision | Parameters |
|---|---|---|
| 100% | 11.6% | Core algorithm, no filters applied |
| 94.7% | 37.9% | Freq.$_{token/modules/language}$:60%, Freq.$_{token/modules/category}$:55%, Min. weight:0 |
| 85.1% | 48.2% | Freq.$_{token/modules/language}$:60%, Freq.$_{token/modules/category}$:25%, Min. weight:0 |
| 39.4% | 80.4% | Freq.$_{token/modules/language}$:70%, Freq.$_{token/modules/category}$:30%, Min. weight:4 |
| 16% | 93.8% | Freq.$_{token/modules/language}$:35%, Freq.$_{token/modules/category}$:25%, Min. weight:4 |

# Outlook



# Summary



| Recall | Precision | Parameters |
|---|---|---|
| 100% | 11.6% | Core algorithm, no filters applied |
| 94.7% | 37.9% | $Freq._{token/modules/language}$:60%, $Freq._{token/modules/category}$:55%, Min. weight:0 |
| 85.1% | 48.2% | $Freq._{token/modules/language}$:60%, $Freq._{token/modules/category}$:25%, Min. weight:0 |
| 39.4% | 80.4% | $Freq._{token/modules/language}$:70%, $Freq._{token/modules/category}$:30%, Min. weight:4 |
| 16% | 93.8% | $Freq._{token/modules/language}$:35%, $Freq._{token/modules/category}$:25%, Min. weight:4 |

@EricBouwers
eric@sig.eu