



OOP 2020, München

Computer, fix' meinen Code!

Machine Learning on Code
in der Praxis

Markus Harrer

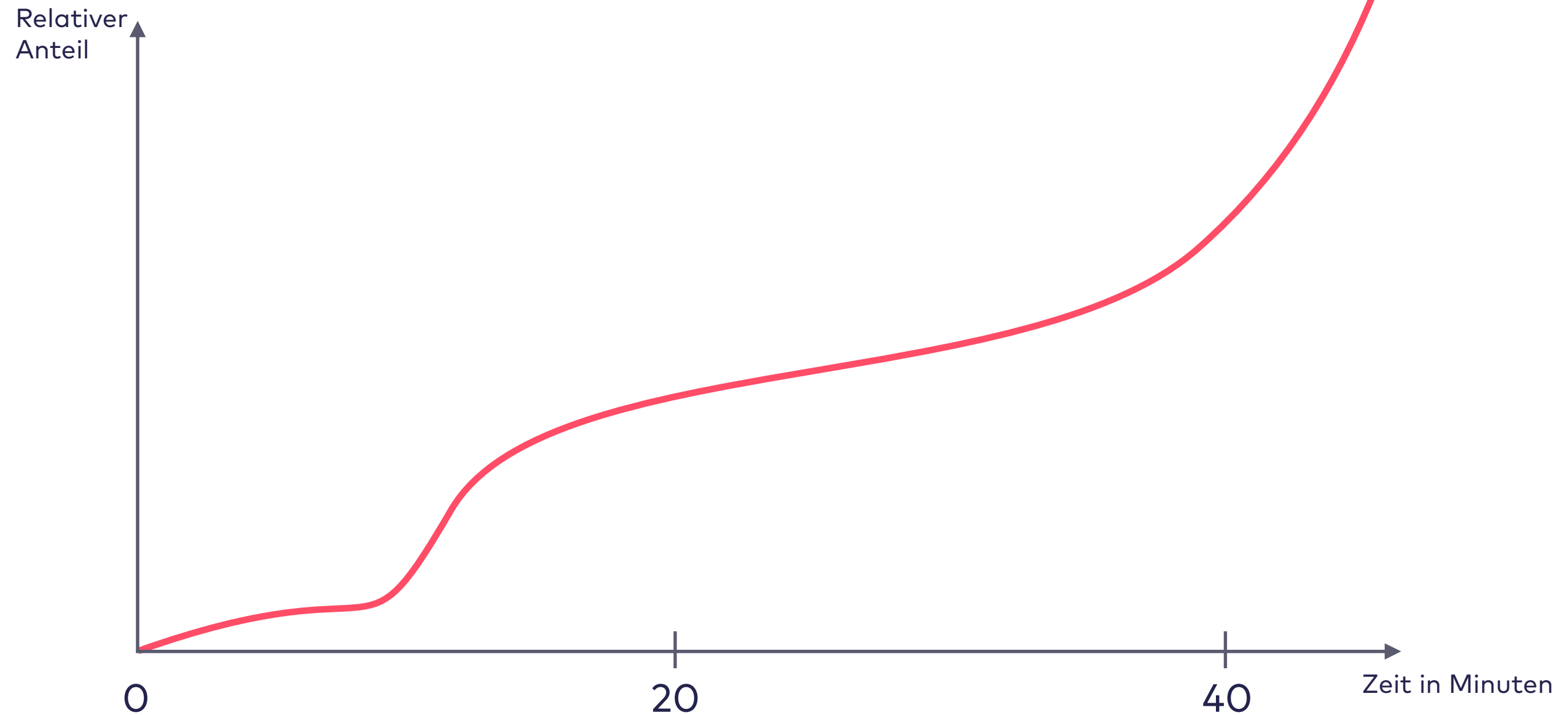
Software Development Analyst

@feststelltaste

INNOQ

Vorab

Anteil von Machine Learning in diesem Talk



Warum dieser Talk?

Aufklären, was heute schon möglich ist!



Warum dieser Talk?

Aufklären, was heute schon möglich ist!

Aus einer anderen Bewertung meiner
Einreichung:

„Wenn er das als Ironie meint, dann würde
ich den Talk sofort akzeptieren!“



Der Schlüsselmoment

Security-Bugfix in der Open-Source-Bibliothek „react-components“

6 yarn.lock	
@@ -6257,9 +6257,9 @@ mississippi@^2.0.0:	
6257	through2 "^2.0.0"
6258	
6259	mixin-deep@^1.2.0:
6260	- version "1.3.1"
6261	- resolved "https://registry.yarnpkg.com/mixin-deep/-/mixin-deep-1.3.1.tgz#a49e7268dce1a0d9698e45326c5626df3543d0fe"
6262	- integrity sha512-8ZIitLHeEgaqEvd51YBXfm4EZSFCX29Jb9K+1AHhDKzReKBQKj3R+7NOF6tjqYi9t4oI8VUfawITJQm86wnXGNQ==
6263	dependencies:
6264	for-in "^1.0.2"
6265	is-extendable "^1.0.1"

6257	through2 "^2.0.0"
6258	
6259	mixin-deep@^1.2.0:
6260	+ version "1.3.2"
6261	+ resolved "https://registry.yarnpkg.com/mixin-deep/-/mixin-deep-1.3.2.tgz#1120b43dc359a785dce65b55b82e257ccf479566"
6262	+ integrity sha512-wRoDn//mXBiJ1H40rqa3vH0toePwSsGb45iInWlTySa+Uu4k3tYUSxa2v1KqAiLtv1rSzaExqS1gtk96A9zvEA==
6263	dependencies:
6264	for-in "^1.0.2"
6265	is-extendable "^1.0.1"

<https://github.com/buildo/react-components/pull/1367/files>

Und wie gefixt? Mit Bots!

Völlig autonome Problembehebung



dependabot bot commented on behalf of github 19 days ago • edited ▾ Contributor +😊 ✎ ⋮

Bumps [mixin-deep](#) from 1.3.1 to 1.3.2.

- ▶ Commits
- ▶ Maintainer changes

compatibility 96%

Dependabot will resolve any conflicts with this PR as long as you don't alter it yourself. You can also trigger a rebase manually by commenting `@dependabot rebase`.

- ▶ Dependabot commands and options

Reviewers

Assignees – assign yourself

Labels

dependencies

Projects

Milestone

Notifications Customize

Unsubscribe

Bumps [mixin-deep](#) from 1.3.1 to 1.3.2.

► Commits

► Maintainer changes

 compatibility 96%

Dependabot will resolve any conflicts with this PR as long as you don't alter it yourself. You can also trigger a rebase manually by commenting `@dependabot rebase`.

► Dependabot commands and options

  **dependabot** bot added the **dependencies** label 19 days ago

  Bump mixin-deep from 1.3.1 to 1.3.2 ...

Verified ✓ cf6ab89

  **dependabot** bot **force-pushed** the `dependabot/npm_and_yarn/mixin-deep-1.3.2` branch from `d795b84` to `cf6ab89` 13 minutes ago

  **mergify** bot merged commit `70ae7c1` into `master` 5 minutes ago
2 checks passed

[View details](#)

[Revert](#)

Assignees – assign yourself



Labels



dependencies

Projects



Milestone



Notifications

[Customize](#)

 Unsubscribe

Mark as unread

1 participant



 Lock conversation

 Bump mixín-deep from 1.3.1 to 1.3.2 ...

Verified ✓ cf6ab89

1 participant



 **dependabot** bot [force-pushed](#) the [dependabot/npm_and_yarn/mixin-deep-1.3.2](#) branch from [d795b84](#) to [cf6ab89](#) 13 minutes ago

 Lock conversation

 **mergify** bot merged commit [70ae7c1](#) into [master](#) 5 minutes ago
2 checks passed

[View details](#)

[Revert](#)



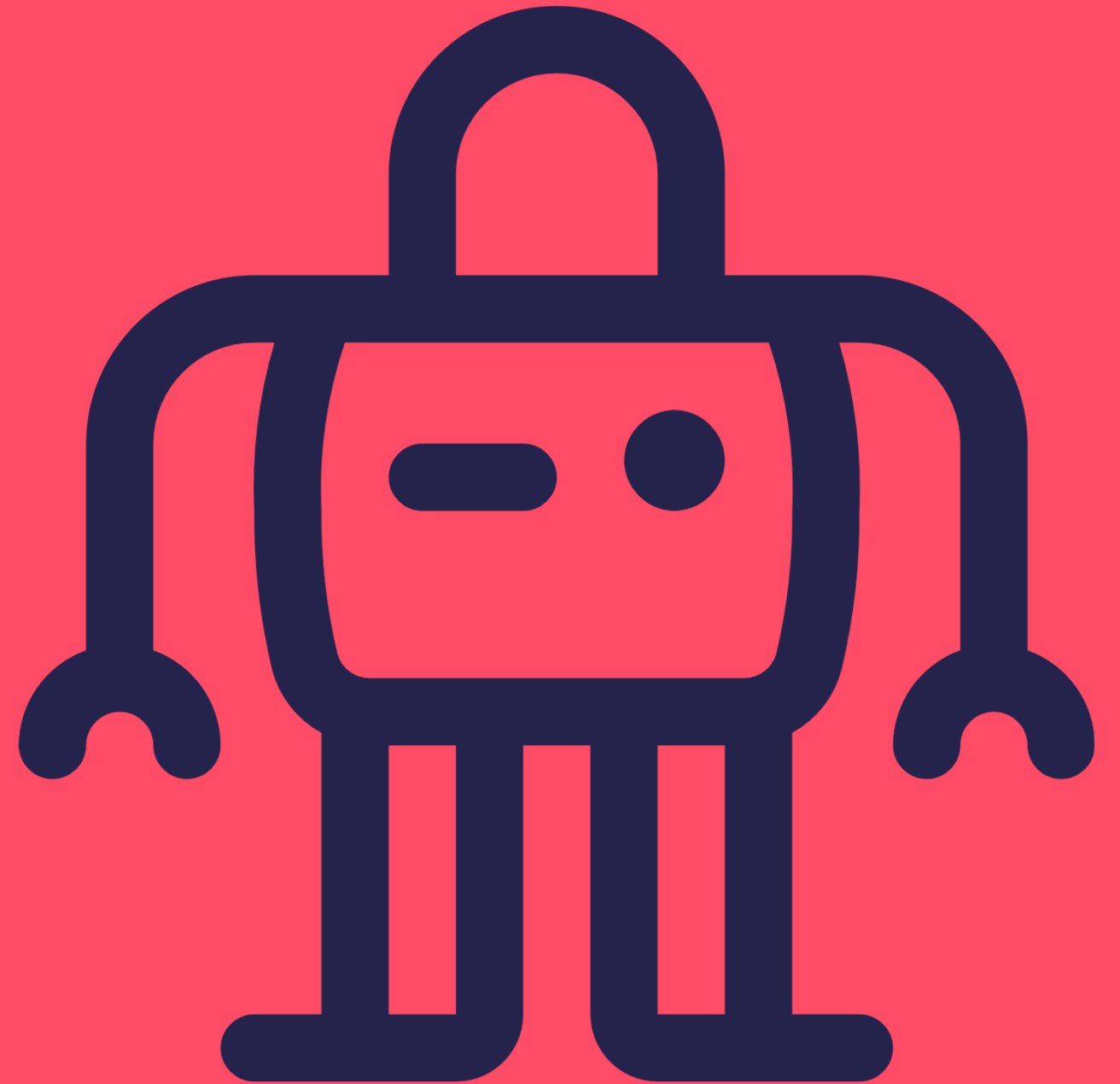
nemobot 5 minutes ago

Member



I, For One, Welcome Our New AI Overlords

Burkhard Neppert, INNOQ



Hintergründe

Eine Choreographie vollzogen von Bots



Dependabot

- Findet Security-Issues
- Kann Patches dafür erstellen



Mergify

- Nimmt Pull-Requests vor, wenn bestimmte Bedingungen vorliegen



Nemobot

- Feiert Pull-Requests mit GIFs (aber unklar, ob echter Bot)



Gabriele Petronella
@gabro27

So this just happened:

- a bot found a vulnerability in a dependency
- a bot sent a PR to fix it
- the CI verified the PR
- a bot merged it
- a bot celebrated the merge with a GIF

github.com/buildo/react-c...

12:43 nachm. · 16. Sep. 2019 · [Tweetbot for Mac](#)

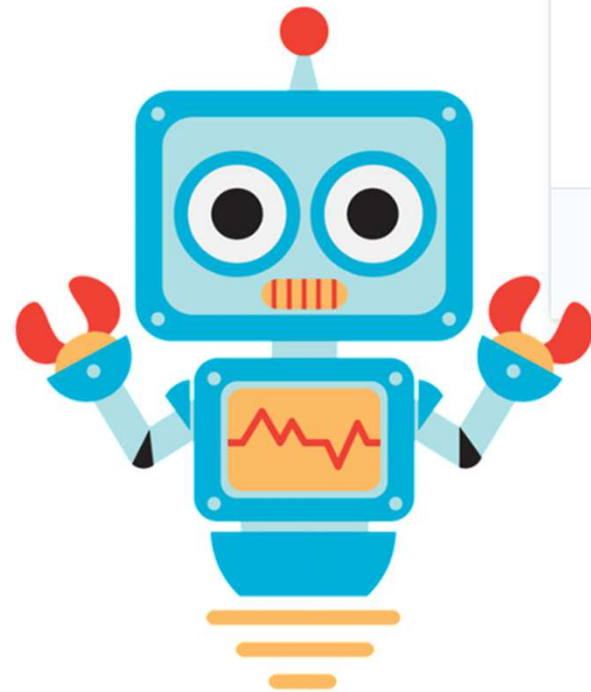
7.000 Retweets **17.760** „Gefällt mir“-Angaben



<https://twitter.com/gabro27/status/1173547934132178944>

Wie geht das?

GitHub Probots: Vorkonfigurierte Apps zur Erweiterung von GitHub



Welcome

Welcomes new users

🔗 740 ★ 102



Pull

Keep your forks up-to-date.



Stale

Close stale Issues and Pull Requests



Delete merged branch

No more manually deleting merged branches, this lovely app does it for you.

🔗 1849 ★ 208



Sentiment Bot

Replies to toxic comments with a maintainer designated reply and a link to the repo's code of conduct

🔗 136 ★ 61



Aber der Reihe nach...

1

Machine Learning

2

Code

3

**Machine Learning
on Code**

4

Zusammenfassung

Aber der Reihe nach...

1

Machine Learning

2

Code

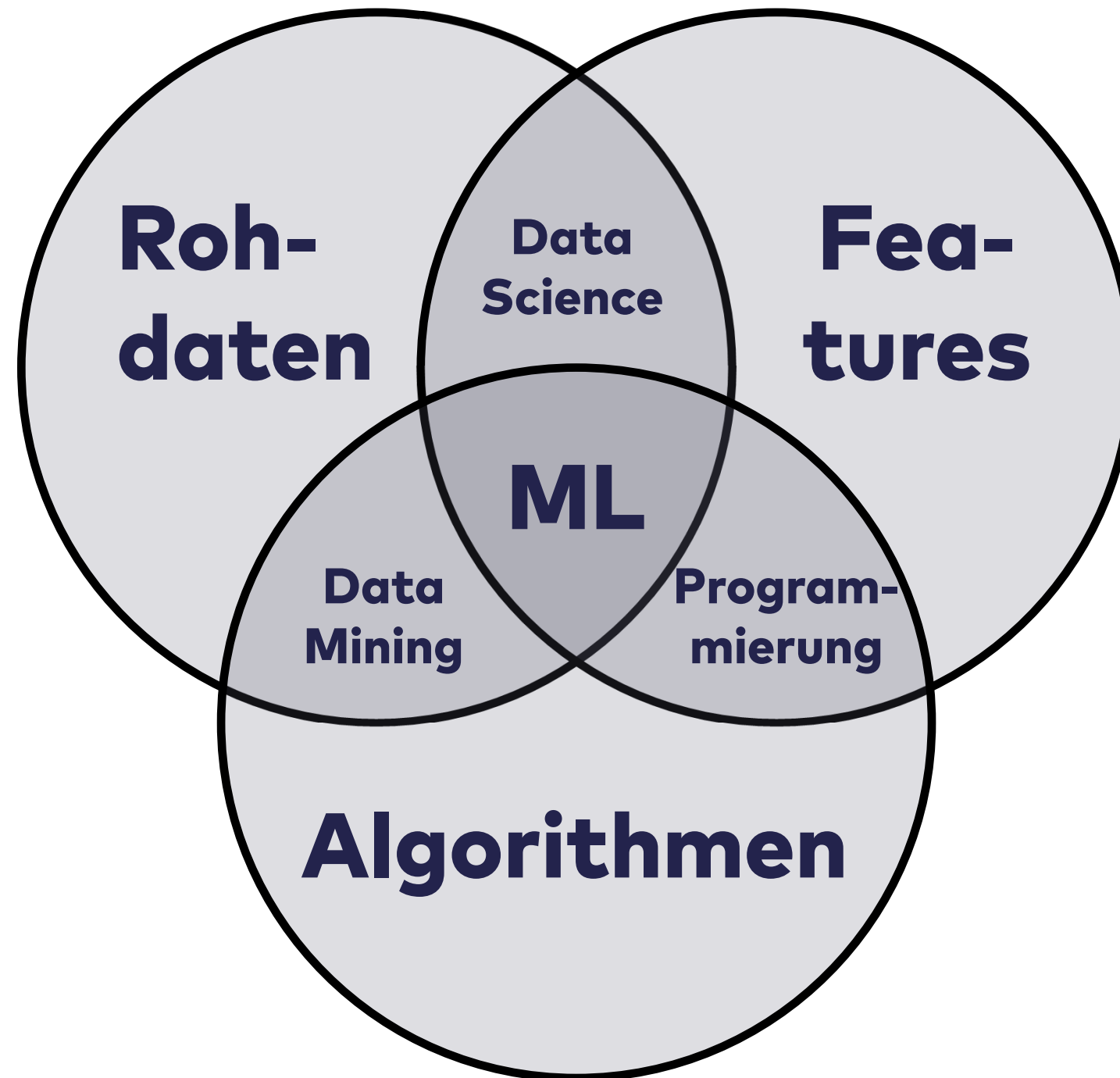
3

**Machine Learning
on Code**

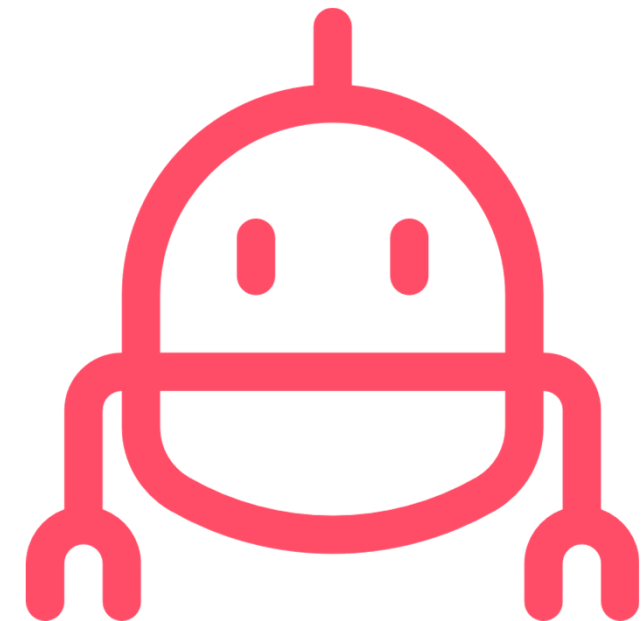
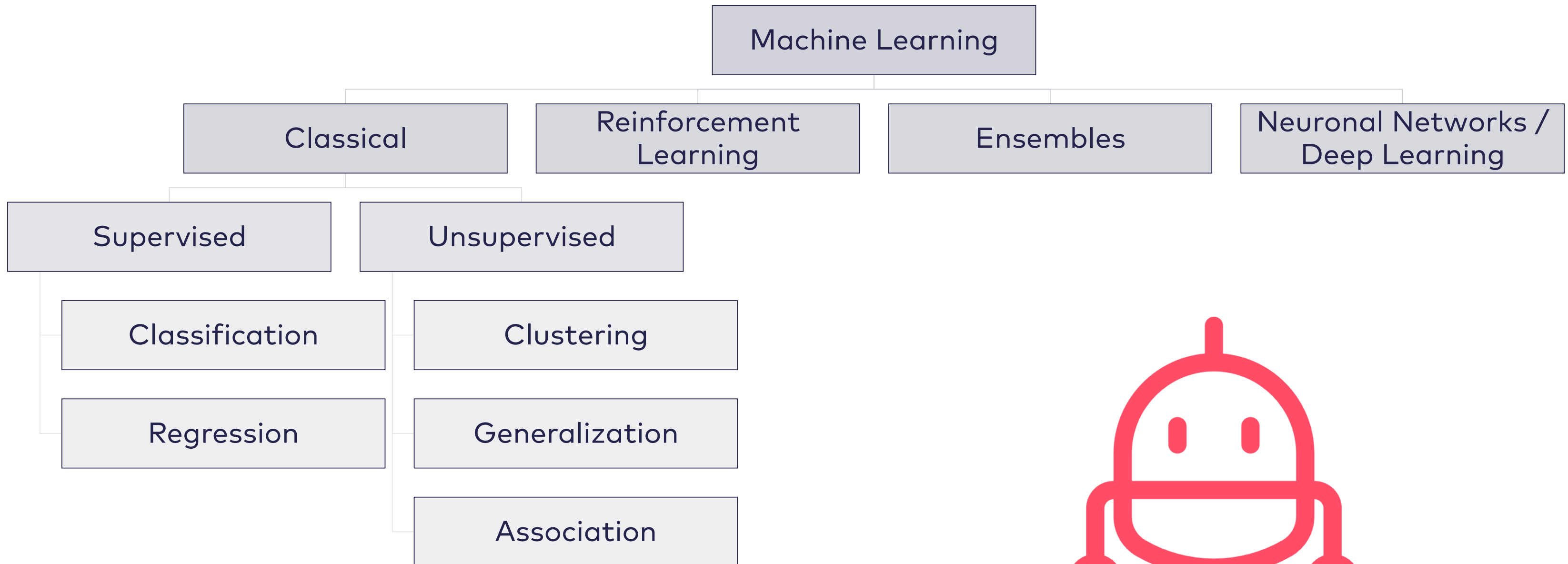
4

Zusammenfassung

Der Kontext von Machine Learning



Machine Learning Kategorien



Aber der Reihe nach...

1

Machine Learning

2

Code

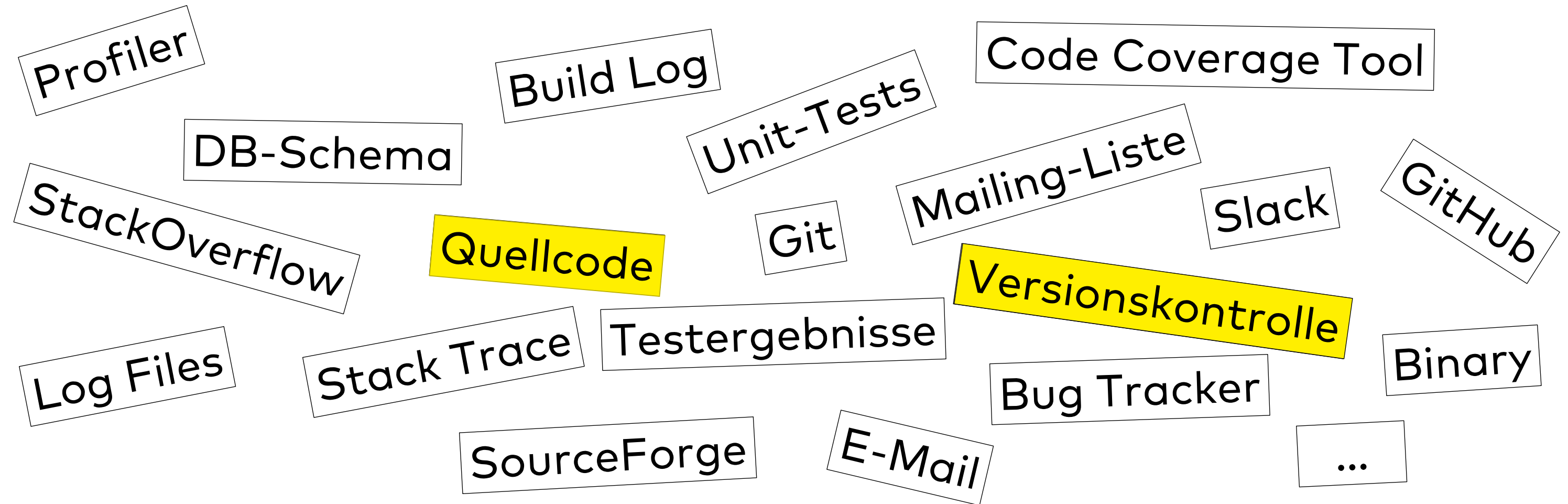
3

**Machine Learning
on Code**

4

Zusammenfassung

Daten in der Softwareentwicklung



Aber der Reihe nach...

1

Machine Learning

2

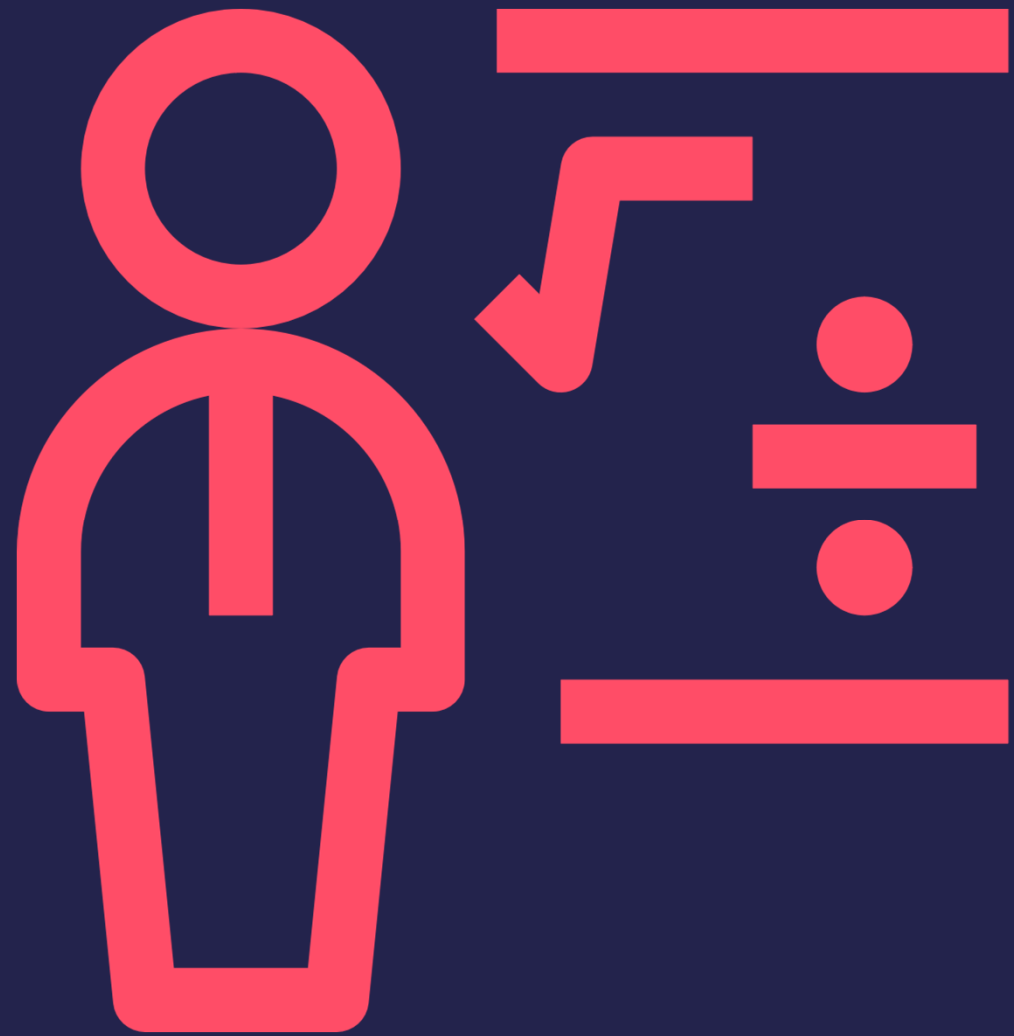
Code

3

**Machine Learning
on Code**

4

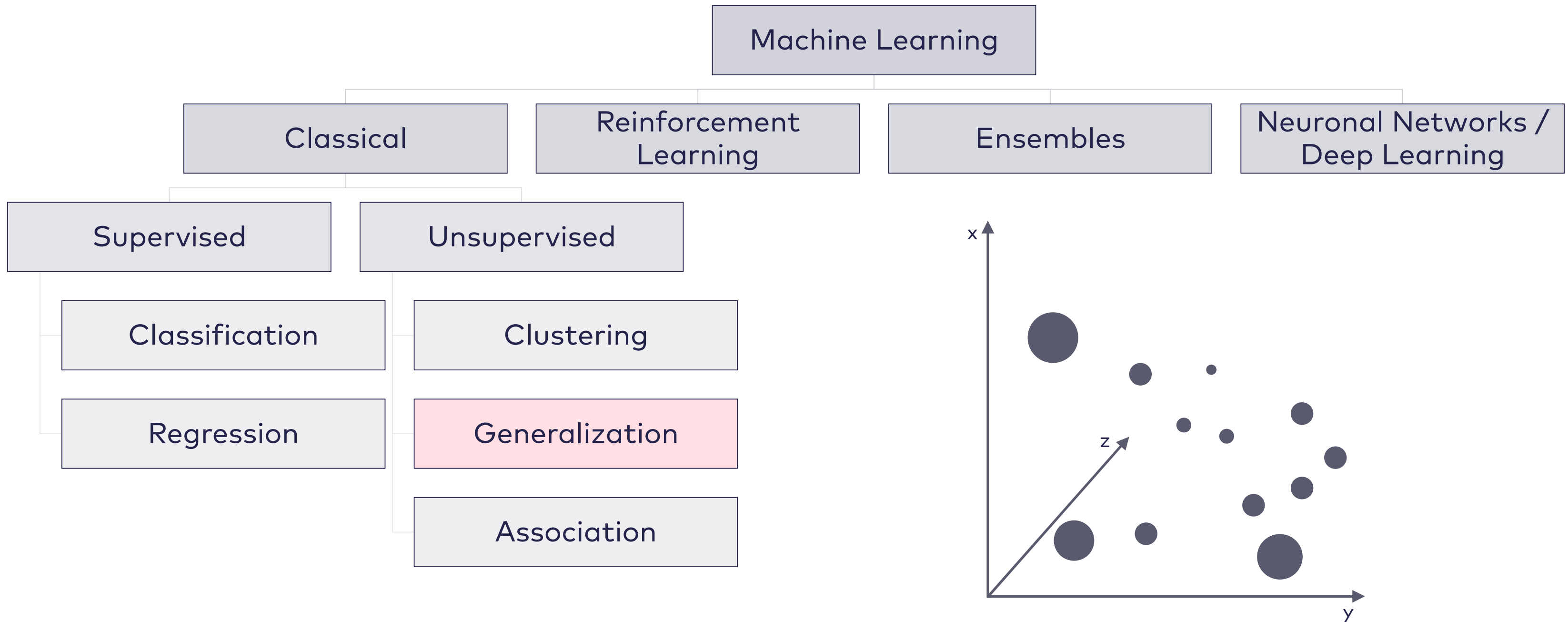
Zusammenfassung



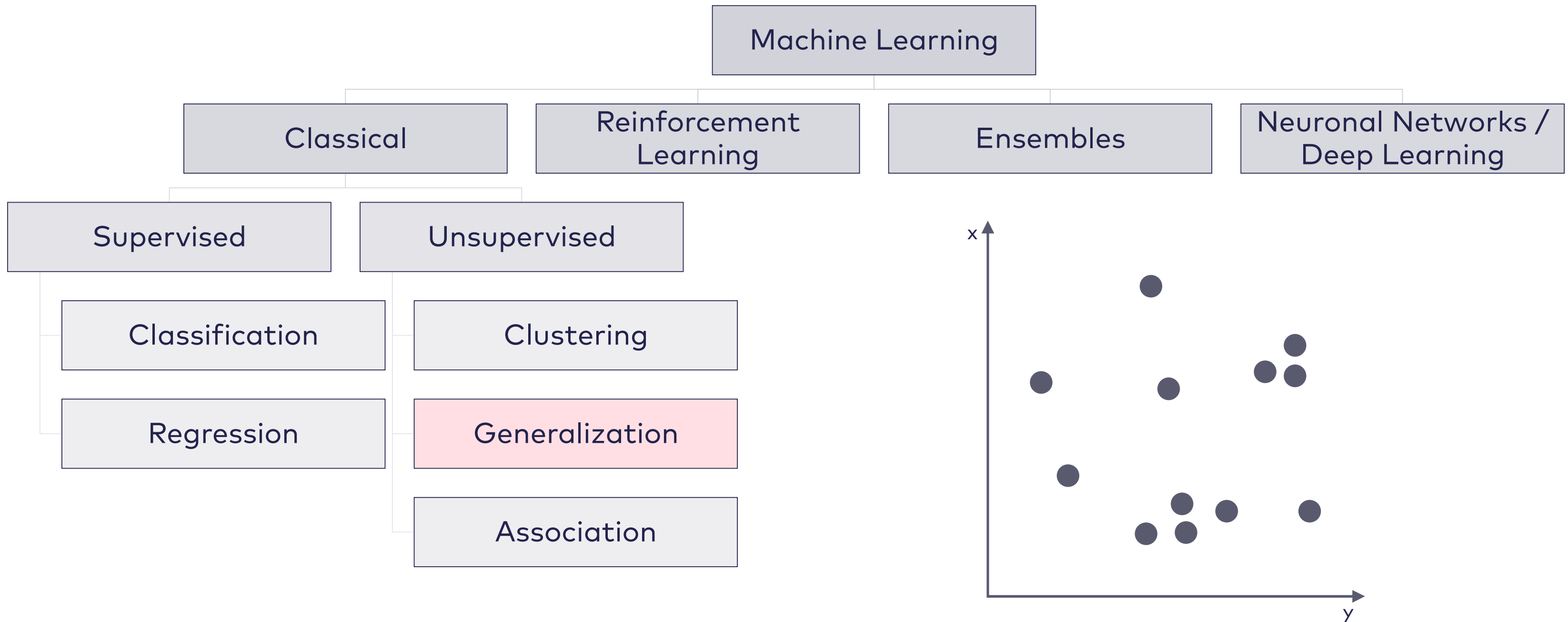
**Können uns
Computer
sagen,
ob wir richtig
entwickeln?**

Generalization

Machine Learning Kategorien

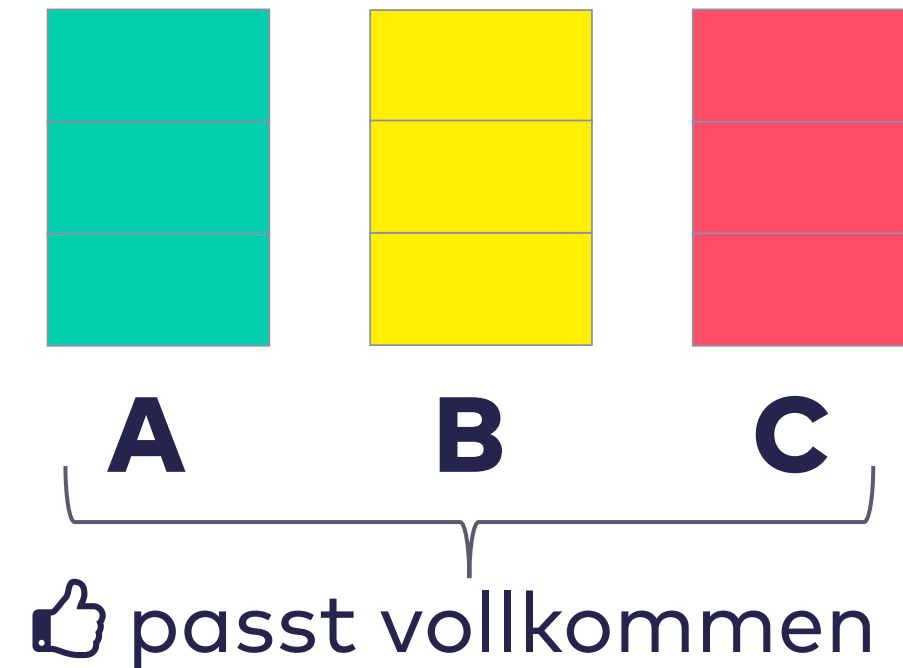
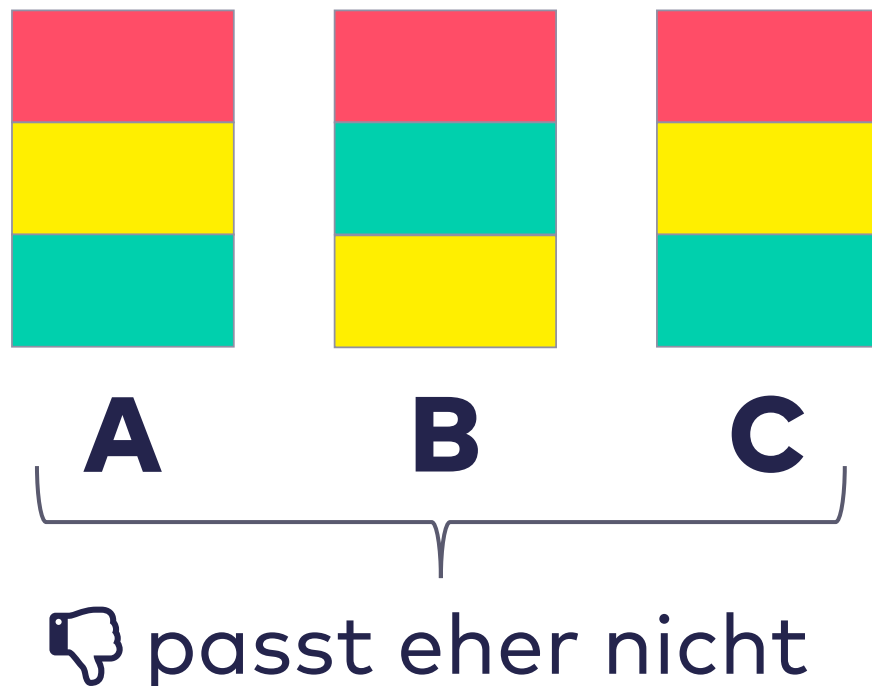


Machine Learning Kategorien



Fragestellung

*„Wie gut passt der fachliche Schnitt zur Arbeitsweise der Entwickler*innen?“*



Legende

A **B** **C** Module

  Änderungen pro Feature

Idee

Heuristik

„Werden **Änderungen** innerhalb einer **Komponente zusammengehörig** vorgenommen?“

- Änderungen => **Commits** aus Versionsverwaltung
- Komponenten => Teil von **Dateipfad**

```
git log --numstat --format=...
```

commit_id	filepath
#59a26	.../todo/Get.java
#59a26	.../todo/New.java
#34af9	.../site/Main.java
...	...

Analyse

Änderungsmatrix mit Commits für jede Datei

	#59a26	#35e25	#34af9	...
.../todo/Get.java	1	1	0	...
.../todo/New.java	1	1	0	...
.../site/Main.java	0	0	1	...
...	...	...	...	...

=> pro Datei ein **Vektor** (=> aber jetzt: reine Mathematik!)

Modellbildung

Ähnlichkeitsberechnung

=> **Cosinus-Ähnlichkeit** zwischen Vektoren (= Quellcode-Dateien)

	.../todo/Get.java	.../todo/New.java	.../site/Main.java	...
.../todo/Get.java	1	0.8	0.3	...
.../todo/New.java	0.8	1	0	...
.../site/Main.java	0.3	0	1	...
...	...	...	...	...

n x n-
dimensionaler
Vektorraum

Generalisierung

Multidimensionale Skalierung reduziert n Dimensionen auf zwei Dimensionen unter Beibehaltung der Abstände

filepath	x	y	komp
.../todo/Get.java	0.14	0.67	todo
.../todo/New.java	0.13	0.70	todo
.../site/Main.java	0.31	0.50	site
...	...	...	...

- + Aus Dateipfad lässt sich Komponente extrahieren
 - Beispiel: .../**todo**/Get.java => **todo**

Visualisierung

Erzeugung einer graphischen Repräsentation als X-Y-Diagramm

Dateien des Softwaresystems

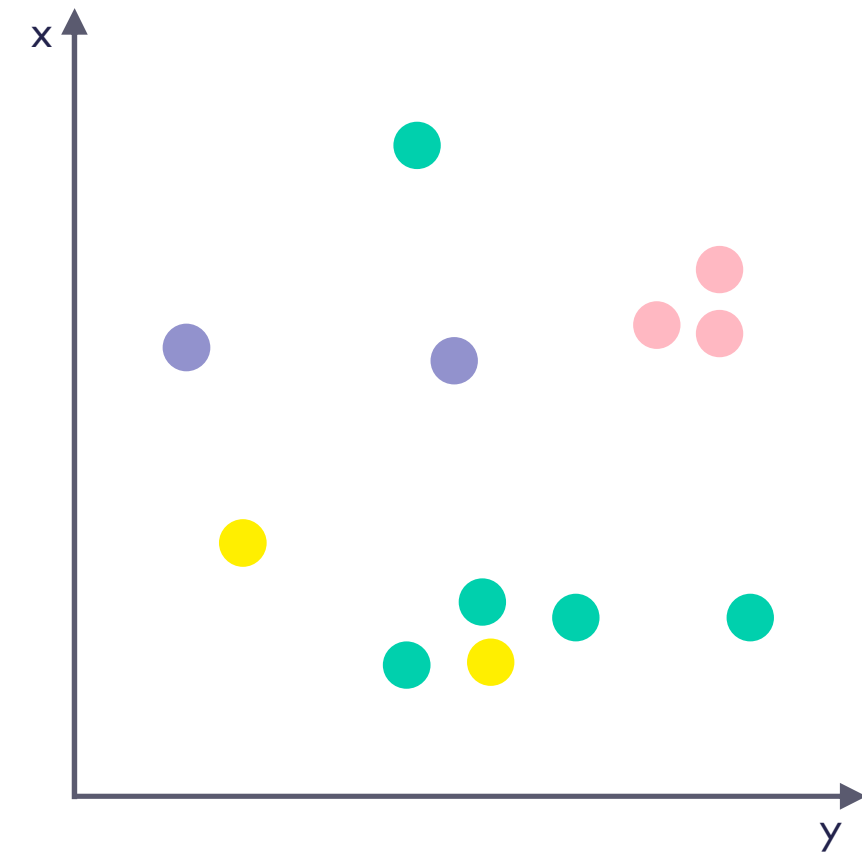
=> **Punkte**

Dateien, die gemeinsam geändert werden

=> **Nähe** der Punkte zueinander

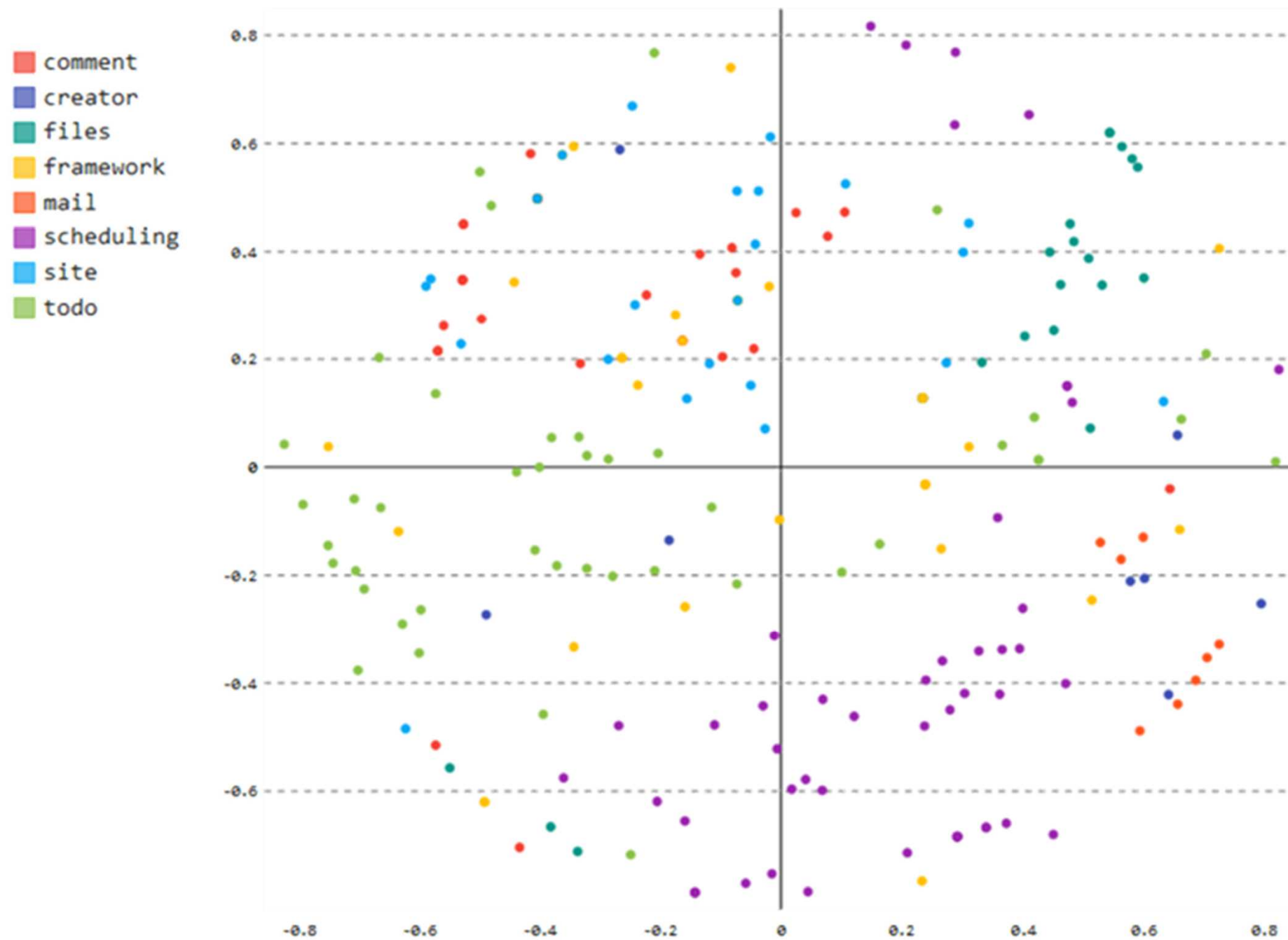
Komponenten des Softwaresystems

=> **Farben** der Punkte

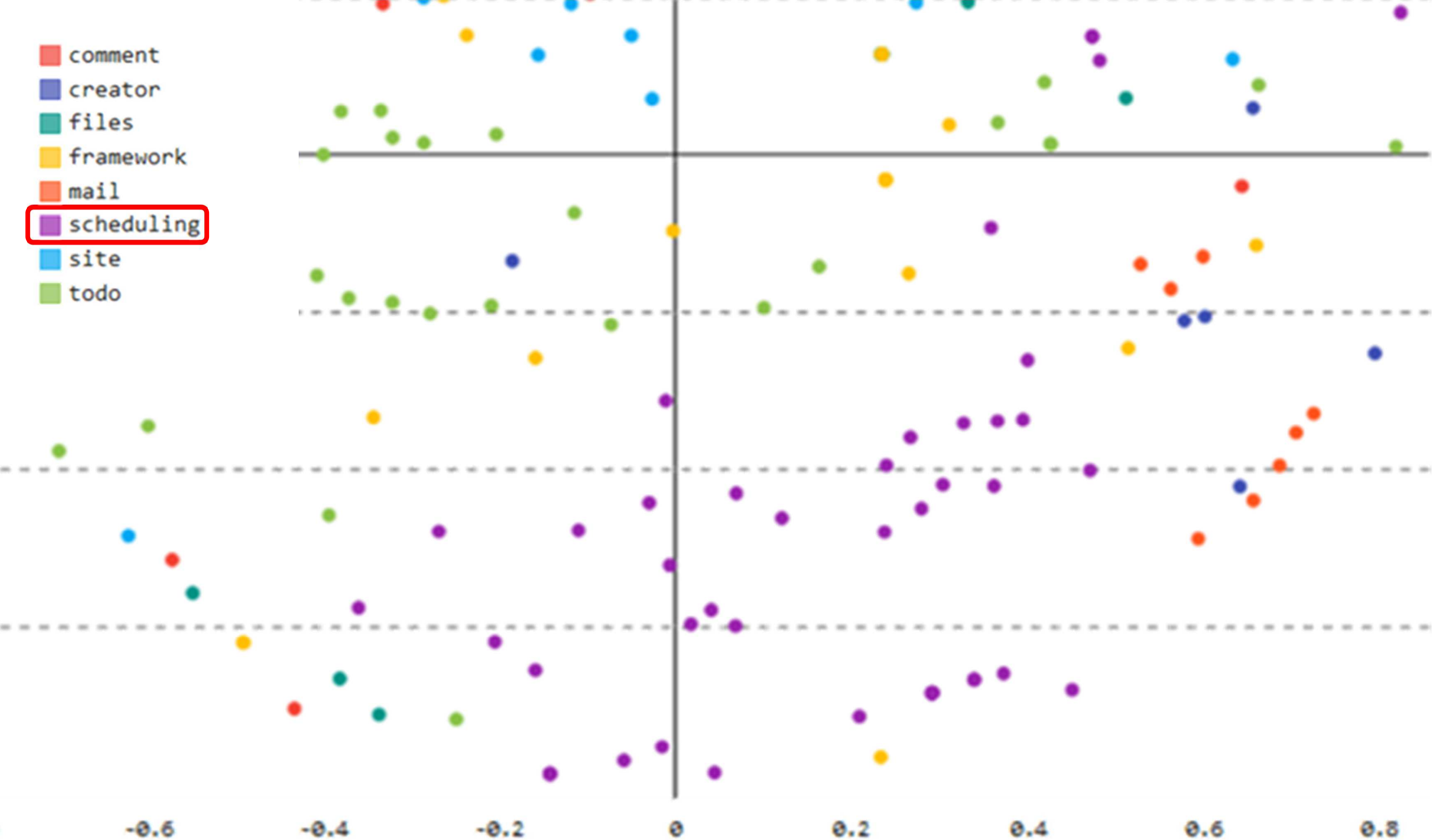


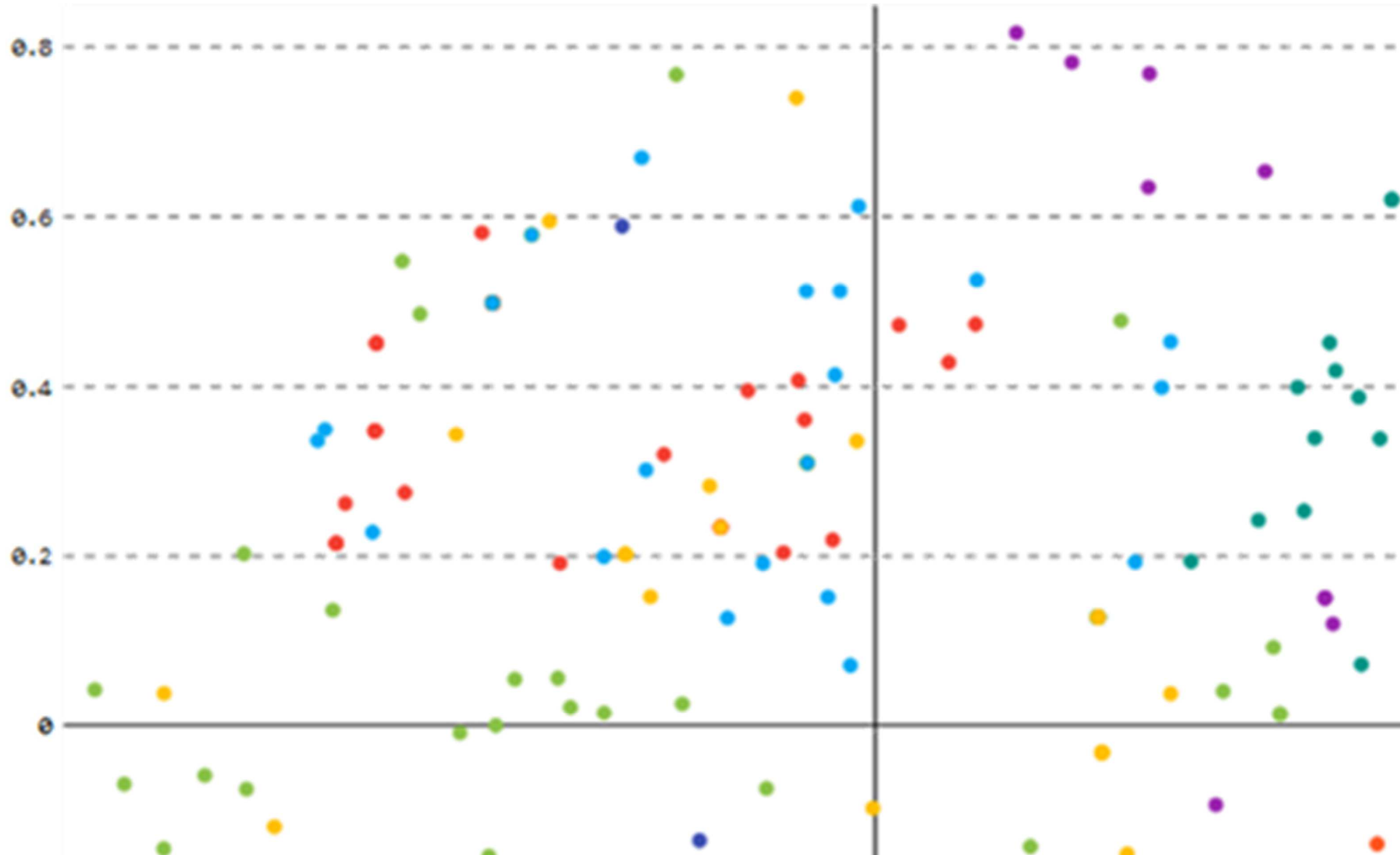
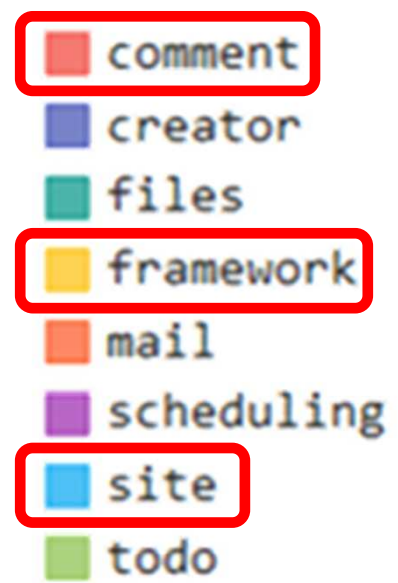
Ergebnis

Änderungskopplung über Modulgrenzen hinweg



- comment
- creator
- files
- framework
- mail
- scheduling
- site
- todo

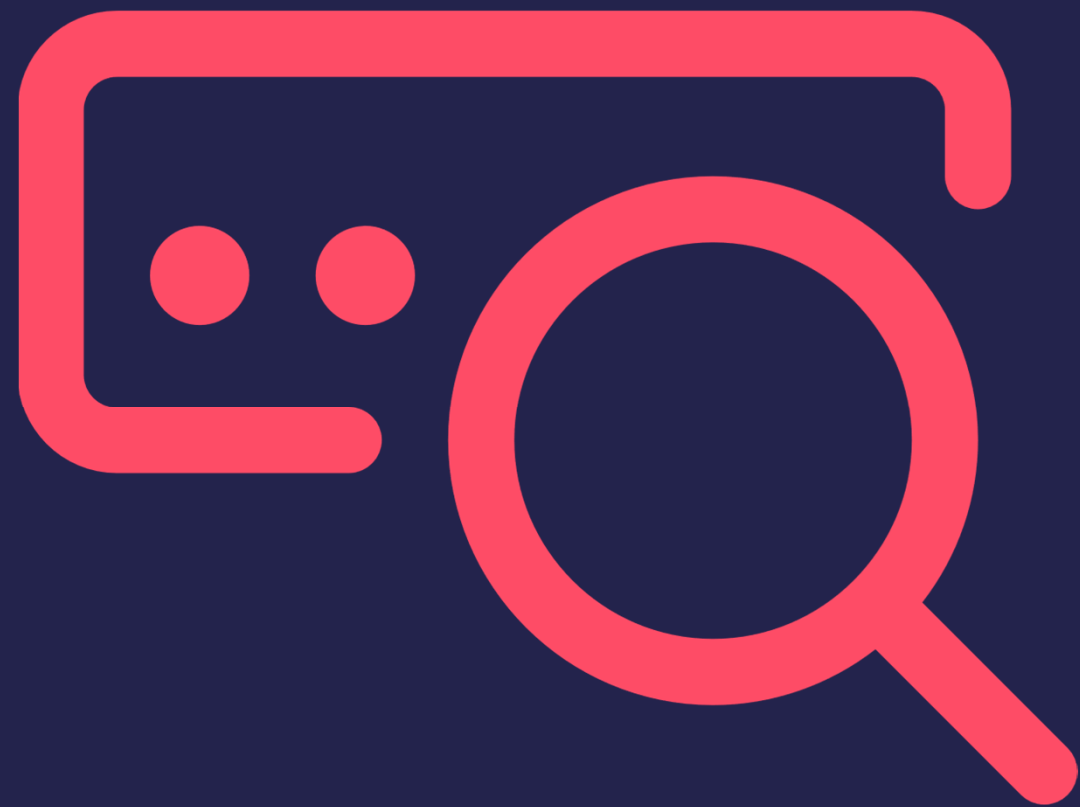






Teils!

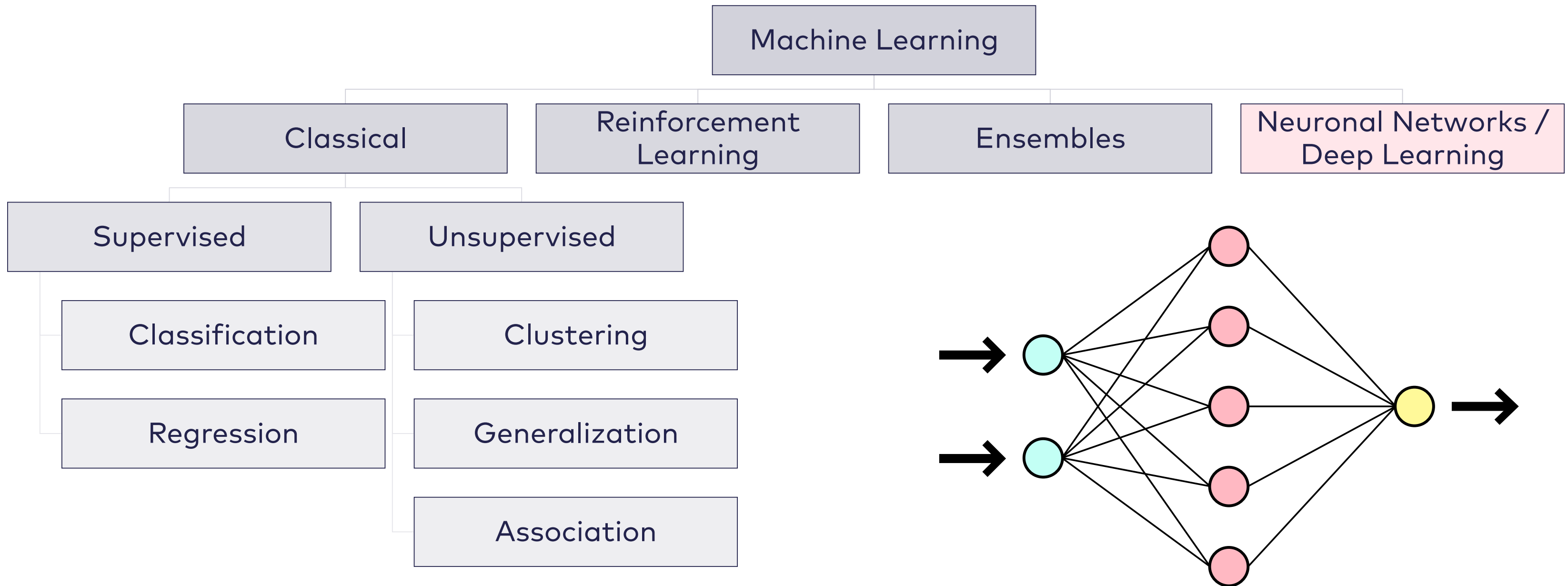
**Können uns
Computer
sagen,
ob wir richtig
entwickeln?**



**Können uns
Computer
sagen, wo wir
passenden Code
finden?**

Neuronal Network

Machine Learning Kategorien



Welcome to our research demo for Semantic Code Search.

Semantic Code Search allows you to find code through meaning instead of keyword matching. **That means the best search results don't necessarily contain the words you searched for.**

This demo is trained on a limited batch of Python code which limits the quality and quantity of results we offer.

Try one of our suggestions

blocks until the server has shut down

return the flow hierarchy of a directed network

create a synthetic dataset and stores it in memory

pretty prints the statistics aggregated by summary

runs tensorflow serving benchmark

GitHub Semantic Code Search

1. Repräsentationen von Code erlernen...

```
===== Example # 30996 =====
```

Original Input:

```
def tearDown self del self user
```

In Tokens über-
führter Code

Original Output:

```
clean up the user instance after each test method.
```

Originaler
Kommentar

***** Predicted Output *****:

```
clean up after each test
```

Zusammengefasster
Kommentar vom
trainierten Modell

GitHub Semantic Code Search

1. ... über code / sentence embeddings und Long short-term memory

```
===== Example # 30996 =====
```

```
Original Input:
```

```
0.32 0.54 0.55 0.72 0.75 0.86 0.85 0.24 0.72 ...
```

Source Code

```
Original Output:
```

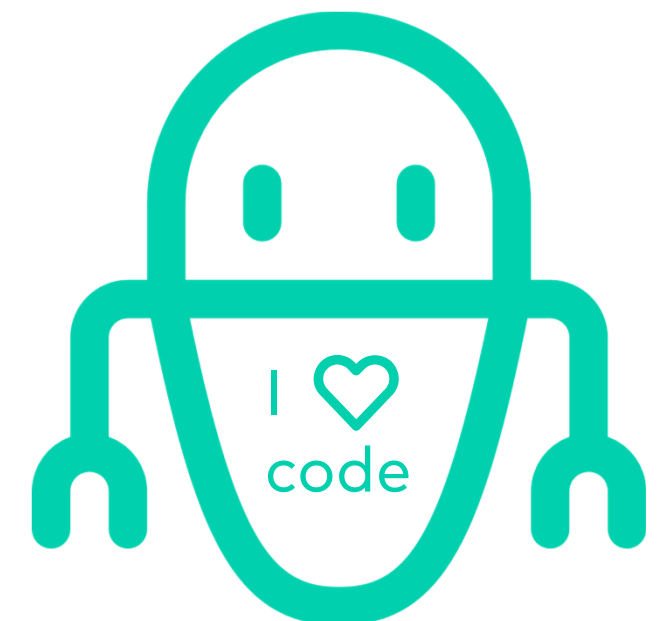
```
0.72 0.52 0.43 0.12 0.54 0.12 0.65 0.23 0.64 ...
```

```
***** Predicted Output *****:
```

```
0.23 0.55 0.55 0.74 0.73 0.88 0.84 0.26 0.74 ...
```

Doc Summary

Möglichst gute
Annäherung
im gleichen
Vektorraum



Code Encoder

GitHub Semantic Code Search

2. Repräsentationen von Text erlernen

`read csv into pandas dataframe`

`load csv or json into pandas dataframe`

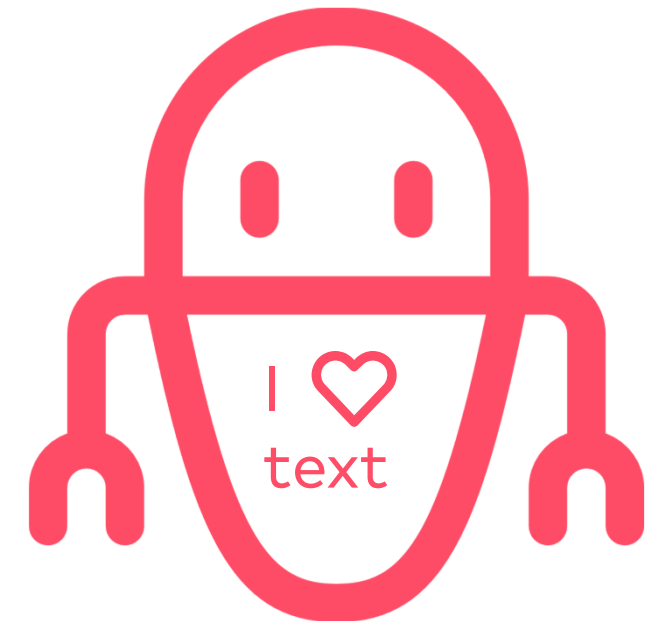
`reads csv files into a raw object` .

→

→

→

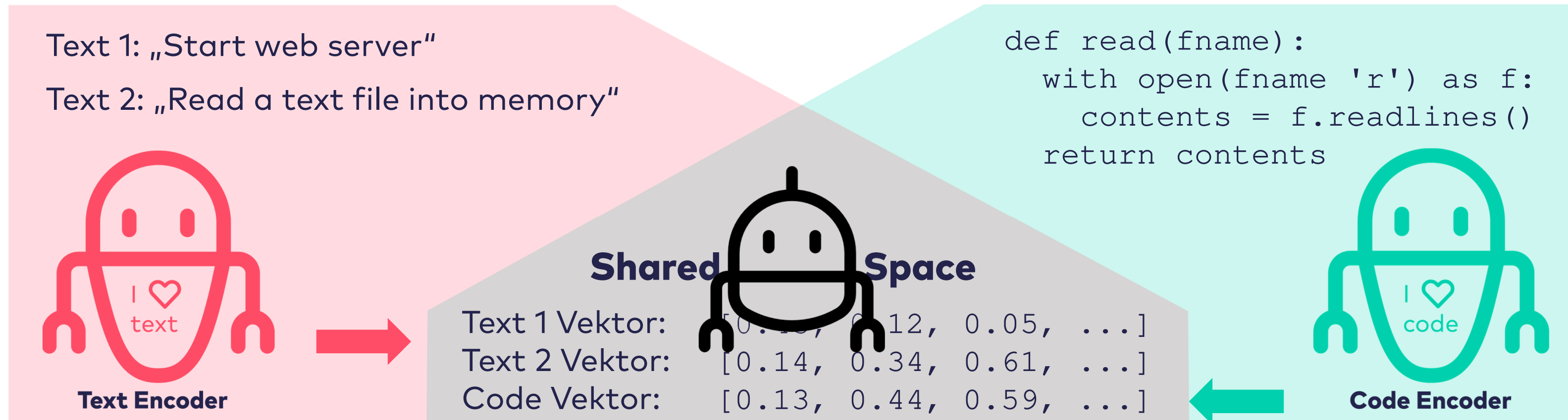
0.72	0.81	...
0.73	0.78	...
0.63	0.93	...



Text Encoder

GitHub Semantic Code Search

3. Code- und Text-Repräsentationen zusammenführen

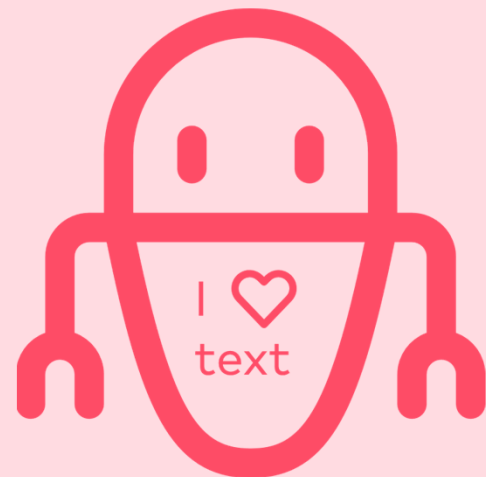


GitHub Semantic Code Search

4. Passenden Code zur Suchanfrage zurückgeben

Text 1: „Start web server“

Text 2: „Read a text file into memory“



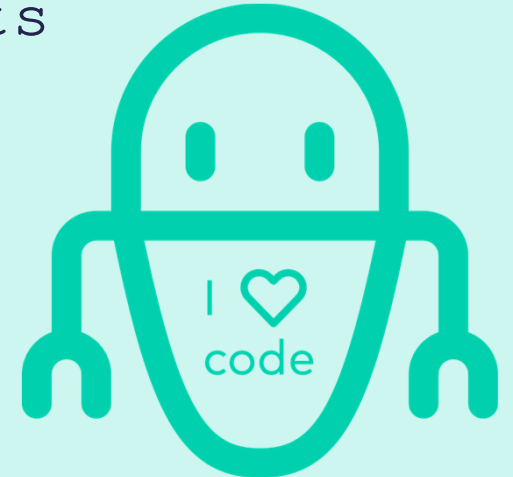
Text Encoder



Shared Vector Space

Text 1 Vektor:	[0.43, 0.12, 0.05, ...]
Text 2 Vektor:	[0.14, 0.34, 0.61, ...]
Code Vektor:	[0.13, 0.44, 0.59, ...]

```
def read(fname):  
    with open(fname 'r') as f:  
        contents = f.readlines()  
    return contents
```



Code Encoder

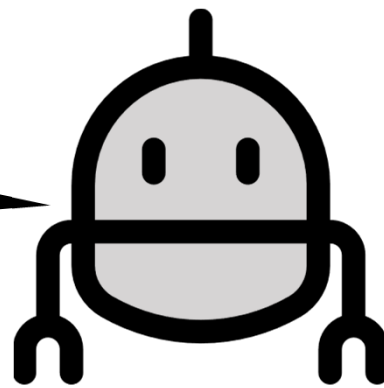


**Berechnung
Cosinus-Distanz**



Text 2 und
Code gehoeren
zusammen!

Nach <https://github.blog/2018-09-18-towards-natural-language-semantic-code-search/>





**Können uns
Computer
sagen, wo wir
passenden Code
finden?**

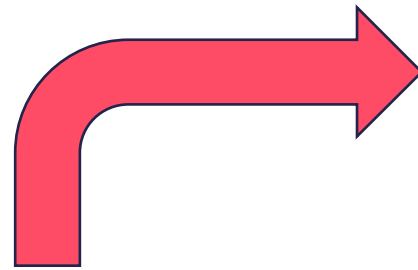
Jo!

Funktional ähnlich: Facebook Aroma

Die Aroma Code Recommendation

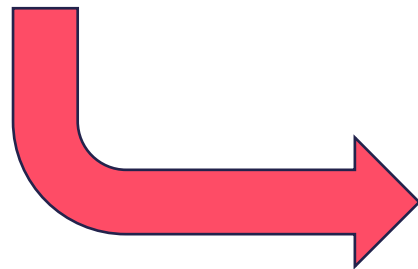
Engine findet ähnlichen Code

zu gegebenen Code



```
1 final BitmapFactory.Options options = new BitmapFactory.Options();
2 options.inSampleSize = 2;
3 // ...
4 Bitmap bmp = BitmapFactory.decodeStream(is, null, options);
```

```
1 Bitmap bitmap = BitmapFactory.decodeStream(input);
```



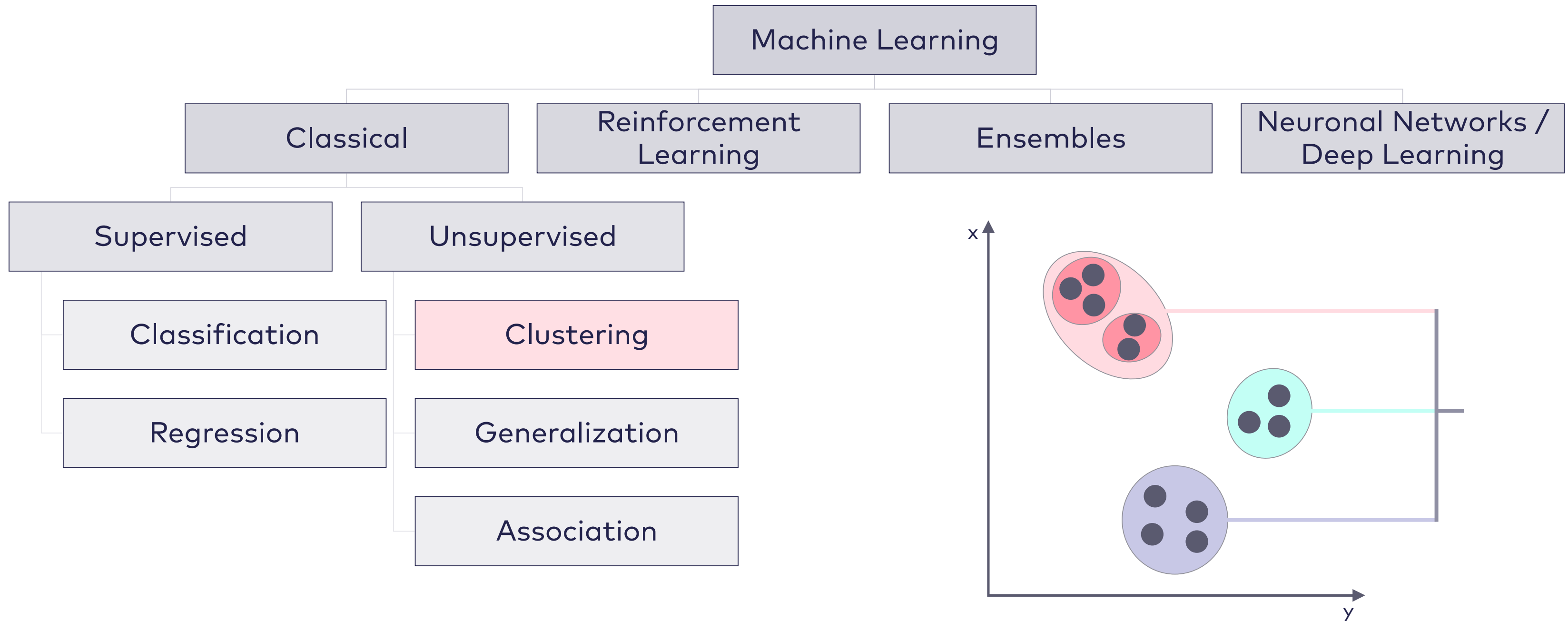
```
1 try {
2     InputStream is = am.open(fileName);
3     image = BitmapFactory.decodeStream(is);
4     is.close();
5 } catch (IOException e) {
6     // ...
7 }
```



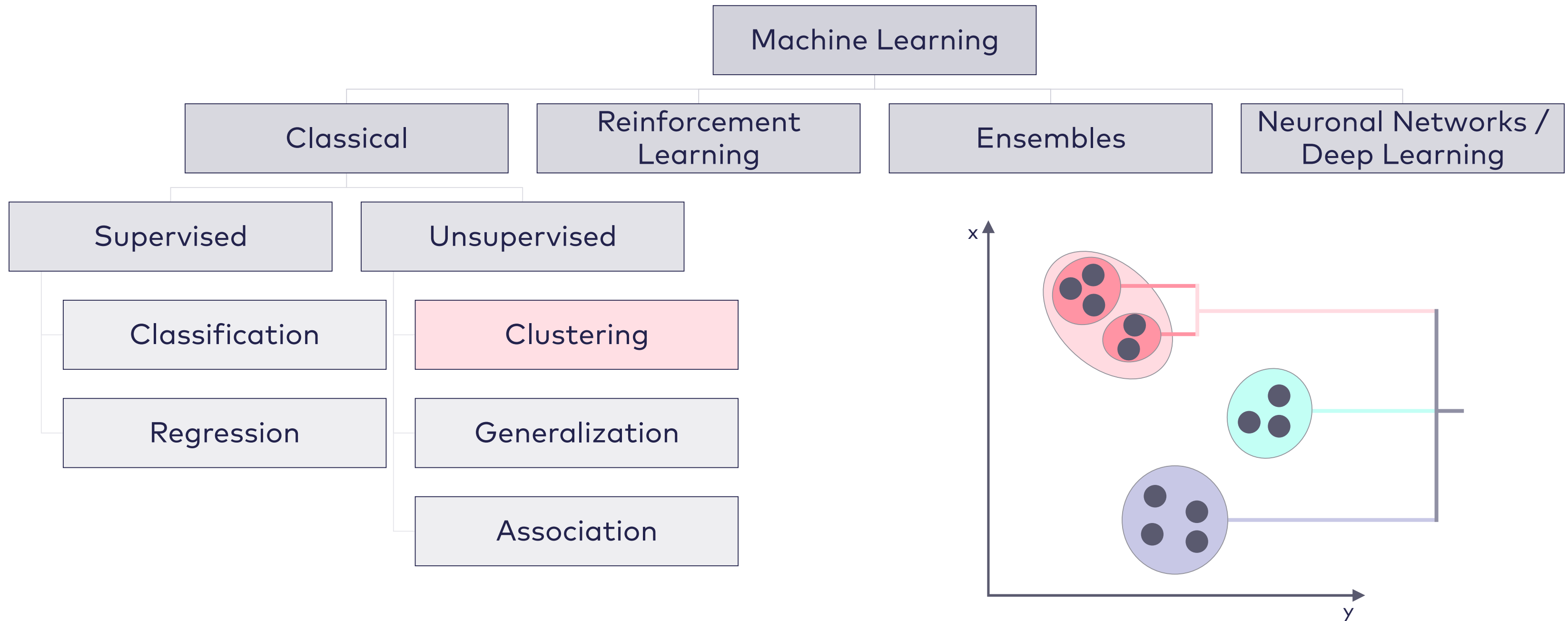
**Können
Computer
Bugs
fixen?**

Clustering

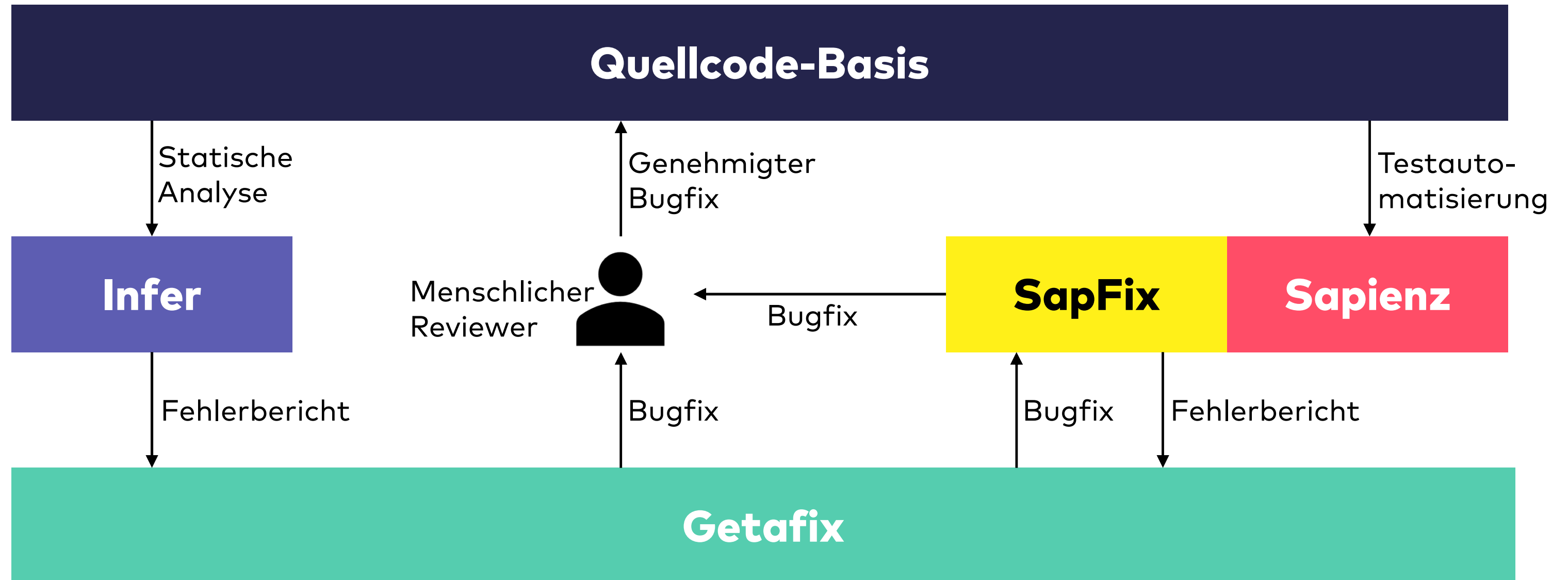
Machine Learning Kategorien



Machine Learning Kategorien



Facebook Getafix: Einsatzkontext



<https://code.fb.com/developer-tools/getafix-how-facebook-tools-learn-to-fix-bugs-automatically/>

Mustererkennung mit Clustering

A: `v.getWidth()` $\longrightarrow$ `v != null ? v.getWidth() : 0`

B: `lst.size()` $\longrightarrow$ `lst != null ? lst.size() : 0`

AB: `h0.h1()` $\longrightarrow$ `h0 != null ? h0.h1() : 0`

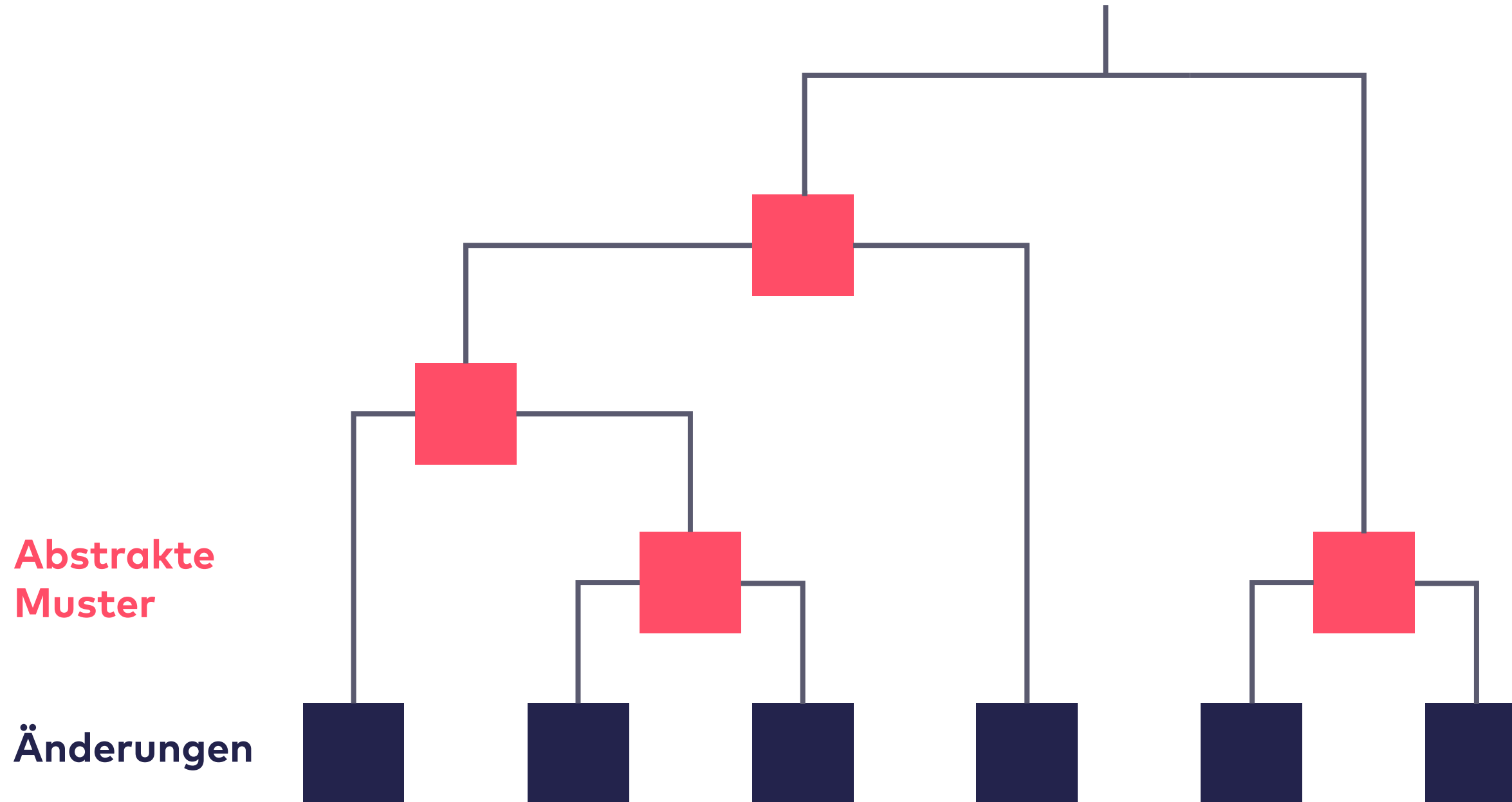
Abstrakte
Muster

Änderungen



Nach einem Beispiel aus dem Konferenzbeitrag „Using Machine Learning for Developer Productivity“

Mustererkennung mit Clustering



Nach einem Beispiel aus dem Konferenzbeitrag „Using Machine Learning for Developer Productivity“

Facebook Getafix: Beispiel

Muster: *h0.h1();* → *if (h0 == null) return; h0.h1();*

```
public void onDestroyView() {  
    mListView.clearListeners();  
    mListView = null;  
}
```



```
public void onDestroyView() {  
    h0.h1();  
    mListView = null;  
}
```



```
public void onDestroyView() {  
    if(mListView == null)  
        return;  
    mListView.clearListeners();  
    mListView = null;  
}
```



```
public void onDestroyView() {  
    if (h0 == null)  
        return;  
    h0.h1();  
    mListView = null;  
}
```

<https://code.fb.com/developer-tools/getafix-how-facebook-tools-learn-to-fix-bugs-automatically/>



**Können
Computer
Bugs
fixen?**

**Immer
öfter!**



**Können
Computer selbst
programmieren?**

Neuronal Network

(nochmal)

CLgen

Deep Learning Program Generator

Generierung lauffähiger
Programme unter
Verwendung von
Deep Learning.

Zumindest zu 50%
nicht von manuell
erstellten Code zu
unterscheiden







**...oder
doch?**

Program Synthesis + Machine Learning

Programming by Example: „Flash Fill“ in Excel

	A	B	C	D	E
1	First Name	Middle Name	Last Name	Combined Name	
2	Mary	Kristy	Mills	Mills, Mary K.	
3	Craig	John	Garland		
4	Andrew		Simpson		
5	Rupert	L	Richards		
6	Joe		Bryant		
7	Jane		Hart		
8	Bob	JD	Bond		
9					
10					

```
if (input[1] == '') then Concat(input[2], Const(",", ""), input[0], Const(".")) else
  Concat(input[2], Const(",", ""), input[0], Const(" "),
    let v = input[1] in Substring(v, AbsPos(v, 0), RegexPos(v, UpperCase, "'", -1)),
    Const("."))
```

<https://www.microsoft.com/en-us/research/publication/automating-string-processing-spreadsheets-using-input-output-examples/> (2011)

```

GenerateStringProgram( $S$ : Set of  $(\sigma, s)$  pairs)
1  $T := \emptyset$ ;
2 foreach  $(\sigma, s) \in S$ 
3    $T := T \cup (\{\sigma\}, \text{GenerateStr}(\sigma, s))$ ;
4  $T := \text{GeneratePartition}(T)$ ;
5  $\tilde{\sigma}' := \{\sigma \mid (\sigma, s) \in S\}$ ;
6 foreach  $(\tilde{\sigma}, \tilde{e}) \in T$ :
7   let  $B[\tilde{\sigma}] := \text{GenerateBoolClassifier}(\tilde{\sigma}, \tilde{\sigma}' - \tilde{\sigma})$ 
8   Let  $(\tilde{\sigma}_1, \tilde{e}_1), \dots, (\tilde{\sigma}_k, \tilde{e}_k)$  be the  $k$  elements in
       $T$  in increasing order of  $\text{Size}(\tilde{e})$ .
9   return Switch( $(B[\tilde{\sigma}_1], \tilde{e}_1), \dots, (B[\tilde{\sigma}_k], \tilde{e}_k)$ );

```

```

GeneratePartition( $S$ : Set of  $(\sigma, s)$  pairs)
1 while exists  $(\tilde{\sigma}, \tilde{e}), (\tilde{\sigma}', \tilde{e}') \in T$  s.t.  $\text{Comp}(\tilde{e}, \tilde{e}')$ 
2   Let  $(\tilde{\sigma}_1, \tilde{e}_1), (\tilde{\sigma}_2, \tilde{e}_2) \in T$  be s.t.  $\text{CS}(\tilde{e}_1, \tilde{e}_2)$ 
      is largest.
3    $T := T - \{(\tilde{\sigma}_1, \tilde{e}_1), (\tilde{\sigma}_2, \tilde{e}_2)\} \cup$ 
       $\{(\tilde{\sigma}_1 \cup \tilde{\sigma}_2, \text{Intersect}(\tilde{e}_1, \tilde{e}_2))\}$ ;
4 return  $T$ ;

```

```

GenerateBoolClassifier( $\tilde{\sigma}_1, \tilde{\sigma}_2$ : Set of inputs)
1  $\tilde{\sigma}'_1 := \tilde{\sigma}_1$ ;  $b := \text{false}$ ;
2 while  $(\tilde{\sigma}'_1 \neq \emptyset)$ 
3    $\text{Old}\tilde{\sigma}'_1 := \tilde{\sigma}'_1$ ;
4    $\tilde{\sigma}'_2 := \tilde{\sigma}_2$ ;  $\tilde{\sigma}''_1 := \tilde{\sigma}'_1$ ;  $d := \text{true}$ ;
5   while  $(\tilde{\sigma}'_2 \neq \emptyset)$ 
6      $\text{Old}\tilde{\sigma}'_2 := \tilde{\sigma}'_2$ ;
7      $\text{Preds} := \{\text{Match}(v_i, r, c), \neg \text{Match}(v_i, r, c) \mid$ 
       $\llbracket \text{Match}(v_i, r, c) \rrbracket \sigma, \sigma \in \tilde{\sigma}_1 \cup \tilde{\sigma}_2\}$ ;
8     Let  $\pi \in \text{Preds}$  be s.t.  $\text{CSP}(\pi, \tilde{\sigma}''_1, \tilde{\sigma}'_2)$ 
      is largest.
9      $d := d \wedge \pi$ ;
10     $\tilde{\sigma}''_1 := \tilde{\sigma}''_1 - \{\sigma_1 \mid \sigma_1 \in \tilde{\sigma}''_1, \neg \llbracket \pi \rrbracket \sigma_1\}$ ;
11     $\tilde{\sigma}'_2 := \tilde{\sigma}'_2 - \{\sigma_2 \mid \sigma_2 \in \tilde{\sigma}'_2, \neg \llbracket \pi \rrbracket \sigma_2\}$ ;
12    if  $(\text{Old}\tilde{\sigma}'_2 = \tilde{\sigma}'_2)$  then FAIL.
13     $\tilde{\sigma}'_1 := \tilde{\sigma}'_1 - \tilde{\sigma}''_1$ ;  $b := b \vee d$ ;
14    if  $(\text{Old}\tilde{\sigma}'_1 = \tilde{\sigma}'_1)$  then FAIL.
15 return  $b$ ;

```

```

GenerateStr( $\sigma$ : Input state,  $s$ : Output string)
1  $\tilde{\eta} := \{0, \dots, \text{Length}(s)\}$ ;
2  $\eta^s := \{0\}$ ;
3  $\eta^t := \{\text{Length}(s)\}$ ;
4  $\tilde{\xi} := \{\langle i, j \rangle \mid 0 \leq i < j \leq \text{Length}(s)\}$ ;
5 Let  $W$  be the mapping that maps edge  $\langle i, j \rangle \in \tilde{\xi}$  to the set
       $\{\text{ConstStr}(s[i : j - 1])\} \cup \text{GenerateSubstring}(\sigma, s[i : j - 1])$ ;
6  $W' := \text{GenerateLoop}(\sigma, s, W)$ ;
7 return  $\text{Dag}(\tilde{\eta}, \eta^s, \eta^t, \tilde{\xi}, W')$ ;

```

```

GenerateLoop( $\sigma$ : Input state,  $s$ : Output string,  $W$ )
1  $W' := W$ ;
2 foreach  $0 \leq k_1, k_2, k_3 < \text{Length}(s)$ :
3    $\tilde{e}_1 := \text{GenerateStr}(\sigma, s[k_1 : k_2])$ ;  $\tilde{e}_2 := \text{GenerateStr}(\sigma, s[k_2 : k_3])$ ;
4    $\tilde{e} := \text{Unify}(\tilde{e}_1, \tilde{e}_2)$ ;
5   if  $(\llbracket \text{Loop}(\lambda w : \tilde{e}) \rrbracket \sigma = \{s[k_1 : k_4]\})$  for some  $k_4$ 
6      $W'(\langle k_1, k_4 \rangle) := W'(\langle k_1, k_4 \rangle) \cup \{\text{Loop}(\lambda w : \tilde{e})\}$ ;
7 return  $W'$ ;

```

```

GenerateSubstring( $\sigma$ : Input state,  $s$ : String)
1 result :=  $\emptyset$ ;
2 foreach  $(i, k)$  s.t.  $s$  is substring of  $\sigma(v_i)$  at position  $k$ 
3    $Y_1 := \text{GeneratePosition}(\sigma(v_i), k)$ ;
4    $Y_2 := \text{GeneratePosition}(\sigma(v_i), k + \text{Length}(s))$ ;
5   result := result  $\cup \{\text{SubStr}(v_i, Y_1, Y_2)\}$ ;
6 return result;

```

```

GeneratePosition( $s$ : String,  $k$ : int)
1 result :=  $\{\text{CPos}(k), \text{CPos}(-(\text{Length}(s) - k))\}$ ;
2 foreach  $r_1 = \text{TokenSeq}(T_1, \dots, T_n)$  matching  $s[k_1 : k - 1]$  for some  $k_1$ :
3   foreach  $r_2 = \text{TokenSeq}(T'_1, \dots, T'_m)$  matching  $s[k : k_2]$  for some  $k_2$ :
4      $r_{12} := \text{TokenSeq}(T_1, \dots, T_n, T'_1, \dots, T'_m)$ ;
5     Let  $c$  be s.t.  $s[k_1 : k_2]$  is the  $c^{\text{th}}$  match for  $r_{12}$  in  $s$ .
6     Let  $c'$  be the total number of matches for  $r_{12}$  in  $s$ .
7      $\tilde{r}_1 := \text{generateRegex}(r_1, s)$ ;
8      $\tilde{r}_2 := \text{generateRegex}(r_2, s)$ ;
9     result := result  $\cup \{\text{Pos}(\tilde{r}_1, \tilde{r}_2, \{c, -(c' - c + 1)\})\}$ ;
10 return result;

```

```

generateRegex( $r$ : Regular Expression,  $s$ : String)
  let  $r$  be of the form  $\text{TokenSeq}(T_1, \dots, T_n)$ .
  return  $\text{TokenSeq}(\text{IParts}_s(T_1), \dots, \text{IParts}_s(T_n))$ ;

```



**Können
Computer selbst
programmieren?**

Ja*

Aber der Reihe nach...

1

Machine Learning

2

Code

3

**Machine Learning
on Code**

4

Zusammenfassung

Zusammenfassung

- **Machine Learning on Code ist noch ein junges Feld**
- **Es gibt viele Forschungsaktivitäten in diesem Bereich**
- **Erste Ansätze von MLonCode nutzen wir bereits im Alltag**

Vielen Dank! Fragen?



www.innoq.com

Markus Harrer

Software Development Analyst

E-Mail: markus.harrer@innoq.com

Twitter: @feststelltaste

Blog: feststelltaste.de

#SoftwareAnalytics
#ArchitectureAsCode
#SoftwareArchitectureReviews
#LegacySystemModernization
#CloudTransformation
#WardleyMapping
#GamificationInSoftwareArchitectureTraining
#Others

innoQ Deutschland GmbH

Krischerstr. 100
40789 Monheim am
Rhein
Germany
+49 2173 3366-0

Ohlauer Str. 43
10999 Berlin
Germany

Ludwigstr. 180E
63067
Offenbach
Germany

Kreuzstr. 16
80331
München
Germany

innoQ Schweiz GmbH

Gewerbestr. 11
CH-6330 Cham
Switzerland
+41 41 743 01 11

Albulastr. 55
8048 Zürich
Switzerland