

*eonum*

EVOLVING SOFTWARE

Ilyas Woo

02.04.2025

# eonum

- Software für Gesundheitswesen
- Digitalisierung von Berichten

# Übersicht

- Problemstellung
- Rakefile
- Unsere Lösung
- Demo
- Ausblick
- Fragen

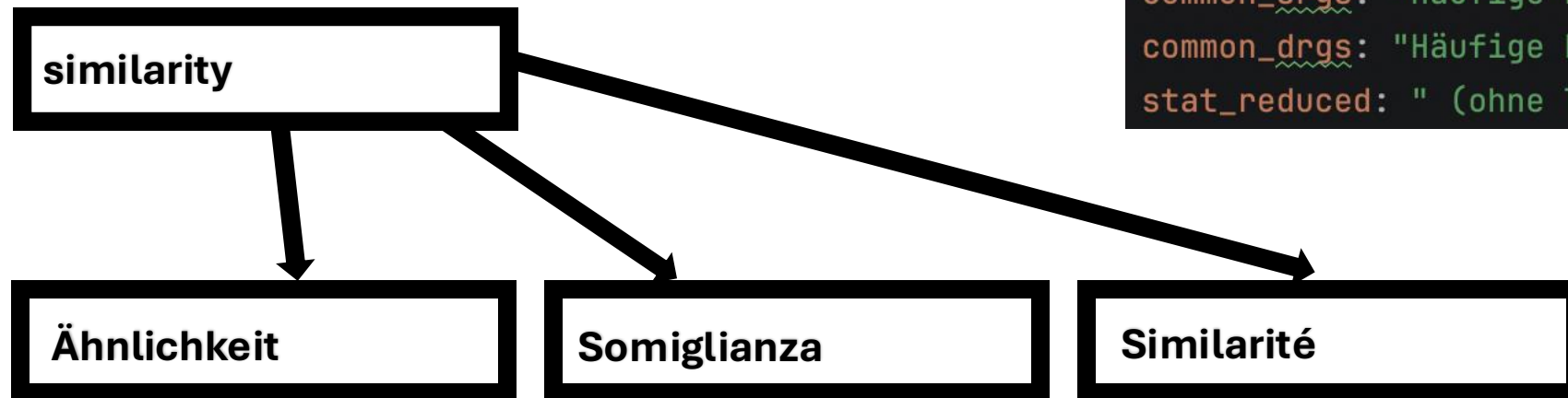
# Problemstellung

- Schweizweit: Deutsch, Französisch, Italienisch
- Jetzt: copy-paste mit deepL
- Ziel: automatische Übersetzungen (ChatGPT)
  - z.B: Diagnose -> Diagnosi (it.)
  - Wichtig: Krebs => ~~Gambero~~  
-> Cancro

# Problemstellung

- YML (YAML) speichert Informationen
- Schlüssel-Wert-Paare (key-value)
- Benennung von Buttons und Labels

```
similarity: "Ähnlichkeit"  
all_cases: "Alle Fälle"  
delta_ecw: "Delta ECW"  
adm_weight: "Aufnahmegewicht"  
num_training: "Fallzahl für DRG-Modell"  
common_pdxs: "Häufige Hauptdiagnosen"  
common_sdxs: "Häufige Nebendiagnosen"  
common_srgs: "Häufige Prozeduren"  
common_drgs: "Häufige DRGs"  
stat_reduced: " (ohne letzte Stelle)"
```



# Problemstellung

- Original in deutsch
- Key-Update Methode existiert
- #FIXME -> übersetzen

```
german: "Deutsch" # FIXME
french: "Französisch" # FIXME
italian: "Italienisch" # FIXME
variables: "Variablen" # FIXME
new_variable: "Neue Variable erstellen" # FIXME
edit_variable: "Variable bearbeiten" # FIXME
back: "Zurück" # FIXME
search: "Suchen" # FIXME
save: "Speichern" # FIXME
name: "Name" # FIXME
description: "Beschreibung" # FIXME
field_name: "Variablenname" # FIXME
```

# Rakefile

- Definiert diverse tasks
- rake 'task\_name'
- Vorteile
  - Leserlich
  - Tests
  - Migrieren von Datenbanken

# RakeFile

```
9 desc 'Translates all values marked with fixme in a given locale-file. Creates a new lo
10 task :translate, [:path] do |t, args|
11     filename = File.basename(args[:path]) #this is used for extracting the name of the f
12     translation_tasks = FixmeFinder.read_locale(args[:path])
13     translation_tasks = TranslateLocales.translate_locales(translation_tasks, filename)
14     LocaleWriter.write_locales(translation_tasks, args[:path])
15     puts "Task completed!"
16 rescue => e
17     puts "Task failed: #{e.message}"
18     raise e
19 end
```



# RakeFile

```
9 desc 'Translates all values marked with fixme in a given locale-file. Creates a new lo
10 task :translate, [:path] do |t, args|
11     filename = File.basename(args[:path]) #this is used for extracting the name of the f
12     translation_tasks = FixmeFinder.read_locale(args[:path])
13     translation_tasks = TranslateLocales.translate_locales(translation_tasks, filename
14     LocaleWriter.write_locales(translation_tasks, args[:path])
15     puts "Task completed:"
16 rescue => e
17     puts "Task failed: #{e.message}"
18     raise e
19 end
```

# RakeFile

```
9 desc 'Translates all values marked with fixme in a given locale-file. Creates a new lo
10 task :translate, [:path] do |t, args|
11     filename = File.basename(args[:path]) #this is used for extracting the name of the f
12     translation_tasks = FixmeFinder.read_locale(args[:path])
13     translation_tasks = TranslateLocales.translate_locales(translation_tasks, filename
14     LocaleWriter.write_locales(translation_tasks, args[:path])
15     puts "Task completed:"
16 rescue => e
17     puts "Task failed: #{e.message}"
18     raise e
19 end
```

# RakeFile

```
9 desc 'Translates all values marked with fixme in a given locale-file. Creates a new lo
10 task :translate, [:path] do |t, args|
11     filename = File.basename(args[:path]) #this is used for extracting the name of the f
12     translation_tasks = FixmeFinder.read_locale(args[:path])
13     translation_tasks = TranslateLocales.translate_locales(translation_tasks, filename
14     LocaleWriter.write_locales(translation_tasks, args[:path])
15     puts "Task completed:"
16 rescue => e
17     puts "Task failed: #{e.message}"
18     raise e
19 end
```

# RakeFile

```
9 desc 'Translates all values marked with fixme in a given locale-file. Creates a new lo
10 task :translate, [:path] do |t, args|
11   filename = File.basename(args[:path]) #this is used for extracting the name of the f
12   translation_tasks = FixmeFinder.read_locale(args[:path])
13   translation_tasks = TranslateLocales.translate_locales(translation_tasks, filename
14   LocaleWriter.write_locales(translation_tasks, args[:path])
15   puts "Task completed:"
16 rescue => e
17   puts "Task failed: #{e.message}"
18   raise e
19 end
```

# RakeFile

```
9 desc 'Translates all values marked with fixme in a given locale-file. Creates a new lo
10 task :translate, [:path] do |t, args|
11     filename = File.basename(args[:path]) #this is used for extracting the name of the f
12     translation_tasks = FixmeFinder.read_locale(args[:path])
13     translation_tasks = TranslateLocales.translate_locales(translation_tasks, filename)
14     LocaleWriter.write_locales(translation_tasks, args[:path])
15     puts "Task completed!"
16 rescue => e
17     puts "Task failed: #{e.message}"
18     raise e
19 end
```

PS PSE\_AI-translation main

Project

db

lib

log

Playground

api\_call\_test

locales

de.yml

en.yml

fr.yml

it.yml

it\_short.yml

.env

.gitignore

fixme\_finder.rb

locale\_writer.rb

openai\_api\_call.rb

Rakefile

translate\_locales.rb

public

storage

test

tmp

gitattributes

.env

README.md

Playground/.../Rakefile

```
1 # frozen_string_literal: true
2
3 require_relative 'openai_api_call'
4 require_relative 'fixme_finder'
5 require_relative 'translate_locales'
6 require_relative 'locale_writer'
7
8
9 desc 'Translates all values marked with fixme in a given locale-file. Creates a new locale-file with the translations.'
10 task :translate, [:path] do |t, args|
11   filename = File.basename(args[:path]) #this is used for extracting the name of the file nwhich contains the Language
12   translation_tasks = FixmeFinder.read_locale(args[:path])
13   translation_tasks = TranslateLocales.translate_locales(translation_tasks, filename)
14   LocaleWriter.write_locales(translation_tasks, args[:path])
15   puts "Task completed!"
16 rescue => e
17   puts "Task failed: #{e.message}"
18   raise e
19 end
20
21
22
```

Terminal

Local

Local (3)

+

-

de.yml

en.yml

fr.yml

it.yml

it\_short.yml

openai\_api\_call.rb

translate\_locales.rb

2 directories, 10 files

ilyas@stud-132-159 api\_call\_test %

PSE\_AI-translation > Playground > api\_call\_test > Rakefile > translate

11:19 LF UTF-8 2 spaces

# Ausblick

- Gems
- Testen

# Fragen