SeemsPhishy

L. Benner, F. Creutz, M. Kiskemper, A. Madhour, M. Vielmeyer, O. Wieder 26.01.2022

Agenda

1. Business Modell
2. Demo
3. Methodik
4. Umsetzung
5. Fazit

1. Business Modell

- Hilfs-Tool für Penetration Testing
- 5 Phases of Penetration Testing:
  a) Planning and reconnaissance
  b) scanning
  c) gaining system access
  d) persistent access
  e) final analysis/report

- Vollständiges Framework für Phishing-Kampagnen
- Automatisch generierte Phishing Mails

→ **Auf Zielorganisation zugeschnitten**

Seems Phishy - Dashboard

192.168.170.31:5000

Calendar Mail Music Network Stuff Studium

Seems Phishy

- Dashboard
- Datasets
- Information Gain
- Text Generation

ADDITIONS

- F.A.Q
- Settings

Dashboard

Home /

Number of Files in Database

12
from 6 Companies

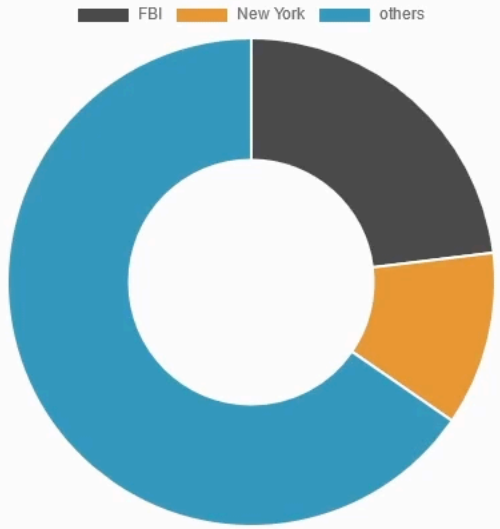
Number of Extracted Keywords

35
in 4 Files

Number of Generated Texts

4
for 3 Companies

Most Common Keywords



Quick Start Guide

Datasets

Information Gain

Text Generation

Phished Companies

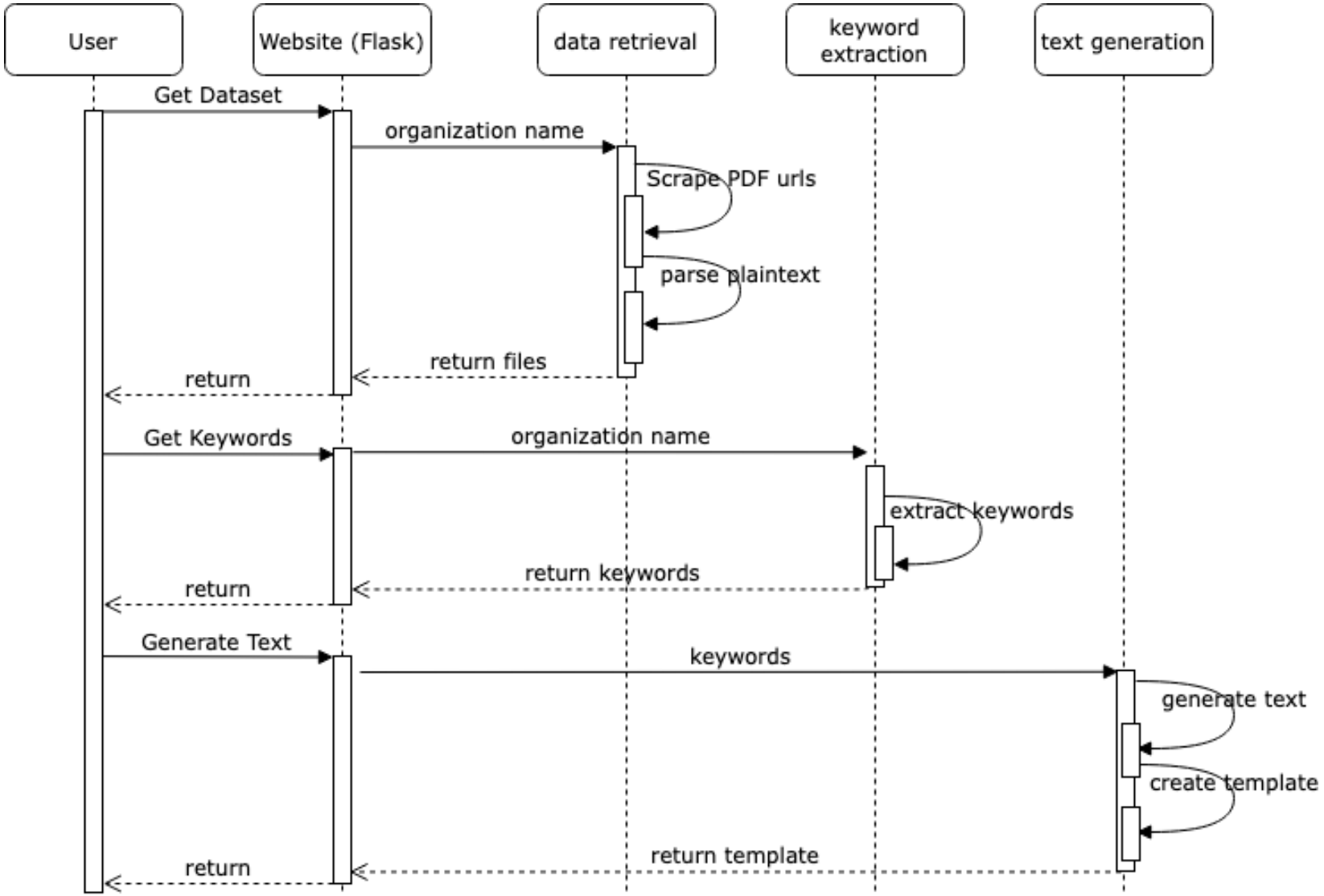
10 entries per page

Search...

entity	status	ner	tfidf	yake_keywords	files
Apple Test 1	In Progress	Not Done	Done	Not Done	2
Apple Test 3	In Progress	Not Done	Done	Not Done	2
HelloFresh	Finished	Not Done	Not Done	Not Done	1
IBM	Finished	Not Done	Not Done	Not Done	1

Warten auf 192.168.170.31...

3. Methodik


User
Website (Flask)
data retrieval
keyword extraction
text generation
Get Dataset
organization name
Scrape PDF urls
parse plaintext
return files
return
Get Keywords
organization name
extract keywords
return keywords
return
Generate Text
keywords
generate text
create template
return template
return

3. Methodik

Vorgehensweise:

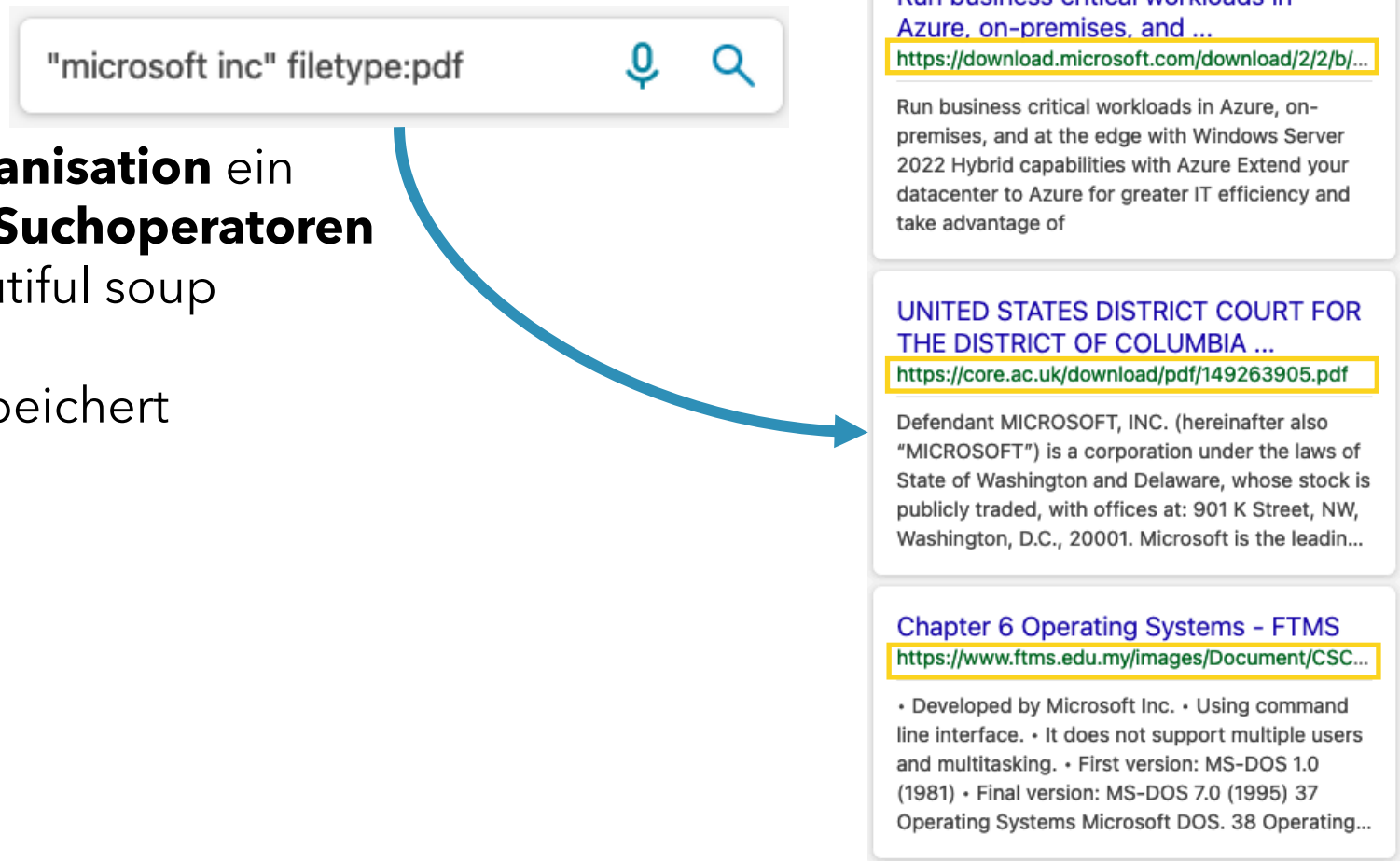
- Parallel
- Weekly Updates für Main-Merge
- Kanban Board über Github

Bereiche/Lanes:

- Data retrieval (Fabian)
- NLP/Keyword extraction (Oliver/Marvin)
- Text generation (Ayman)
- Datenbank (Marius)
- GUI/Flask (Lukas)

4. Umsetzung

- User gibt **Namen der Organisation** ein
- Abfrage mit **verfeinerten Suchoperatoren**
- PDF-URLs werden via beautiful soup **gescraped**
- Anschließend lokal abgespeichert

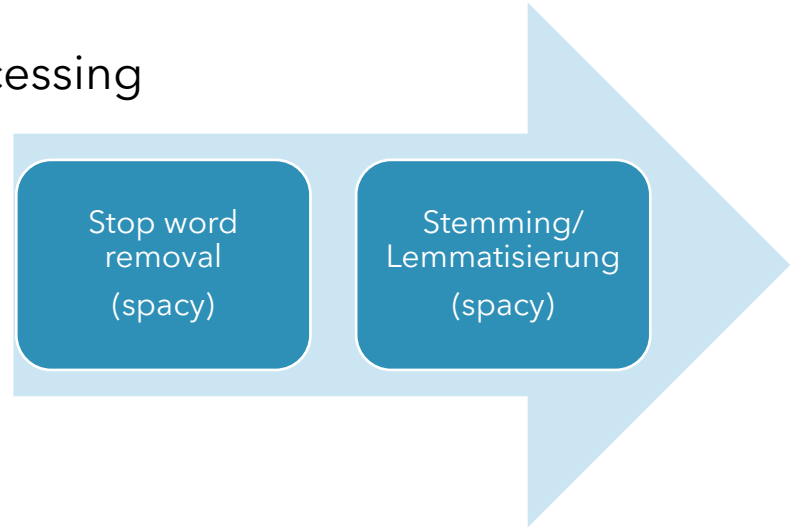


PDF-Miner:

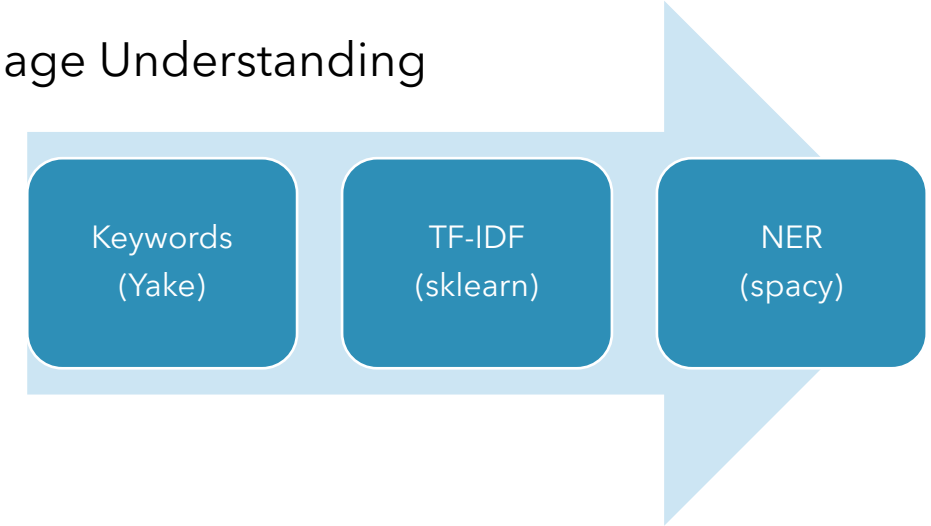
- Text Extraction
- Automatic Layout Analysis

4. Umsetzung

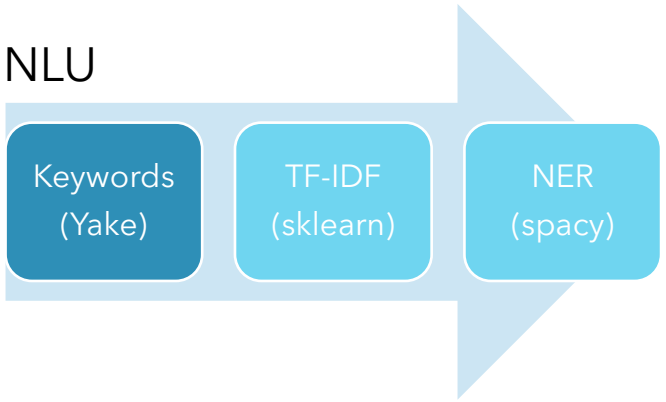
Pre-Processing



Natural Language Understanding



4. Umsetzung



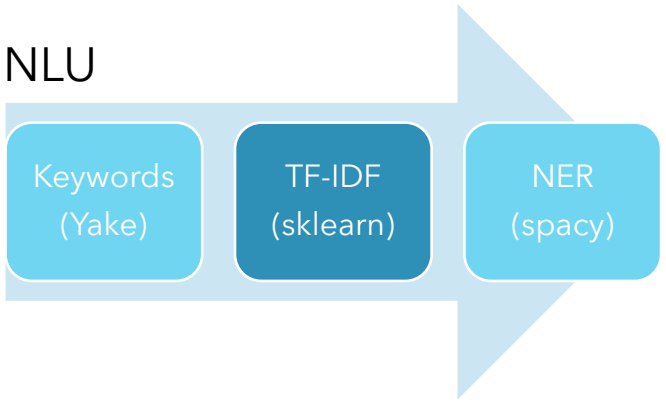
„The Apple identity is a seal of approval and a promise of excellence. When you are authorized or certified in your area of business or expertise, you also represent Apple. By following these guidelines, you reap the benefits of the Apple identity and contribute to its strength."

Erwartete Keywords: **[Apple, seal of approval, promise of excellence, guidelines, Apple identity]**

Yake: [Apple, seal of approval, promise of excellence, guidelines, Apple identity]

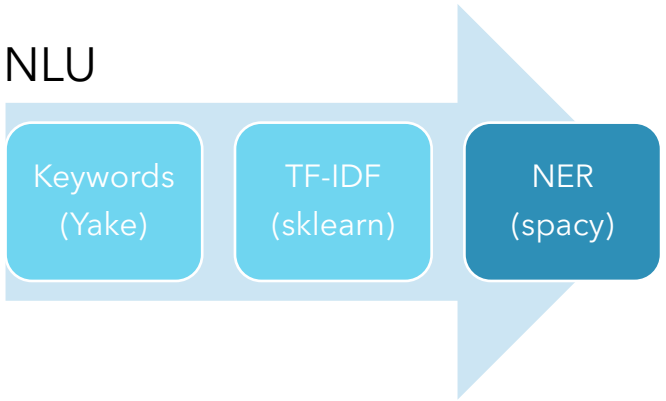
Rake-Nltk: [Apple, seal of approval, promise of excellence, guidelines, Apple identity]

4. Umsetzung



- Keywords von Sklearn:
  - ['product', 'apple', 'inc', 'customer', 'device', 'establish', 'computer', 'provide', 'company', 'cellphone', 'television', 'order', 'executive', 'creation', 'quality', 'transition', 'range', 'assurance', 'retail', 'seek', 'chain', 'summary', 'wearable', 'well', 'wide‘]
  - Geringerer Aufwand (spacy ca. 300 Zeilen)

4. Umsetzung



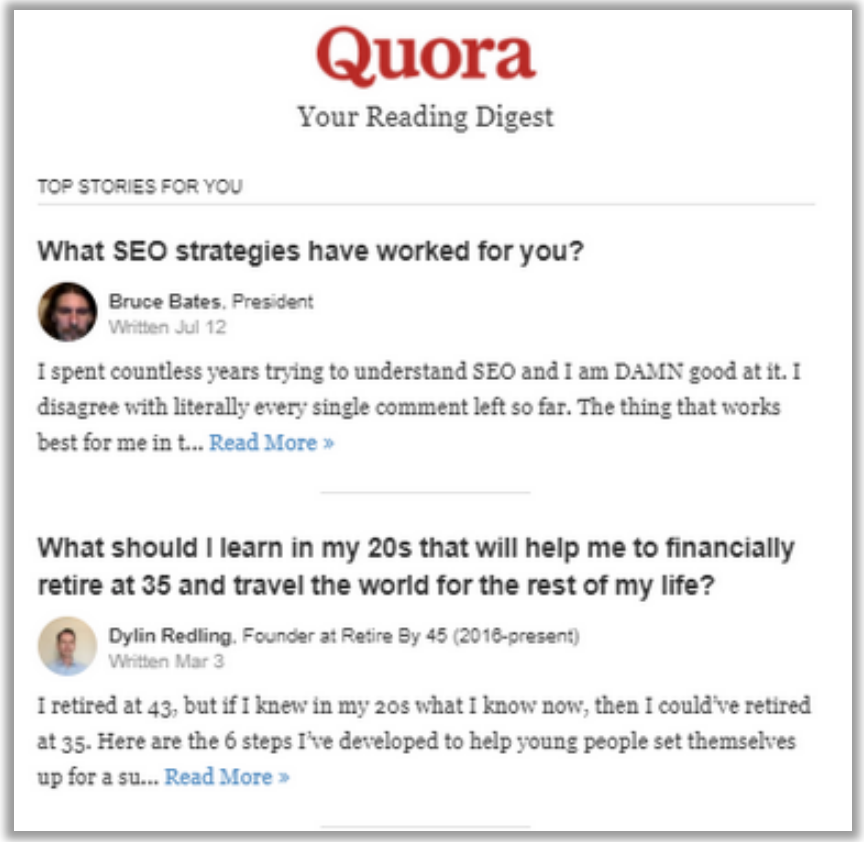
- Keine Empfehlung!
- Learnings:
  - Für den Use-case schwierig einsetzbar
  - Yake erkennt bereits viele wichtige Entities

4. Umsetzung

Ziel:
- Automatisch generierte Phishing-email
- Angepasst an Zielperson
- Interessanter und (halbwegs) realistischer Text
- call-to-action

→ **Newsletter nach dem Quora Format**

Format:
1. Frage
   - Generische Frage, erzeugt **Aufmerksamkeit**
2. Author
   - Vertrauenserweckendes Bild, sowie Erfahrung
3. Antwort
   - Storytelling, **weckt Interesse** des Lesers
   - **Call-to-action** in Form von “Read More”



4. Umsetzung

Challenges:

1. Frage und dazugehörige Antwort generieren
   → Fragenmodell → Input für Antwortenmodell?

2. Text generieren unter vorbedingung (Conditional generation)
   → GPT-2 auf use-case anpassen (**finetuning**)

3. Kein geeigneter Quora-Datensatz
   → **Explain it like I'm 5** (ELI5) Datensatz

4. Umsetzung

Question: Why do TV shows hide logos ?

ELI5 Answer: nothing is free. In most cases, it is a prop for the show, but because apple did NOT pay them for the [product placement](URL0), the show isn't going to give it away. In other cases, apple may not want their brand used in association with that media.

Full Text Keywords: Why do TV **shows** hide **logos** ?: nothing is free. In most cases, it is a prop for the show, but because apple did not pay them for the **product placement**, the show …

Training input:
<|BOS|><|SEP|>*shows,logos,product placement*<|SEP|> Why do TV shows hide logos ?: nothing is free…<|EOS|>

→ **Huggingface Trainer Library**

4. Umsetzung

Try it out!
huggingface.co/Madhour/gpt2-eli5

Input: <|BOS|><|SEP|>Apple,hacking<|SEP|>

Output: Why would Apple try and hack into the iPhone and/or iPad if it can?: There was a good bit of debate about what it was to steal information. The actual reason wasn't quite clear…

Limitationen:

- Grundlegender Bias aufgrund der Daten
  → Kein Domain knowledge
  → Germany = Beer oder Nazi

- Entities werden nicht zuverlässig in den Text mit eingebaut
  → NER im Preprocessing

4. Umsetzung

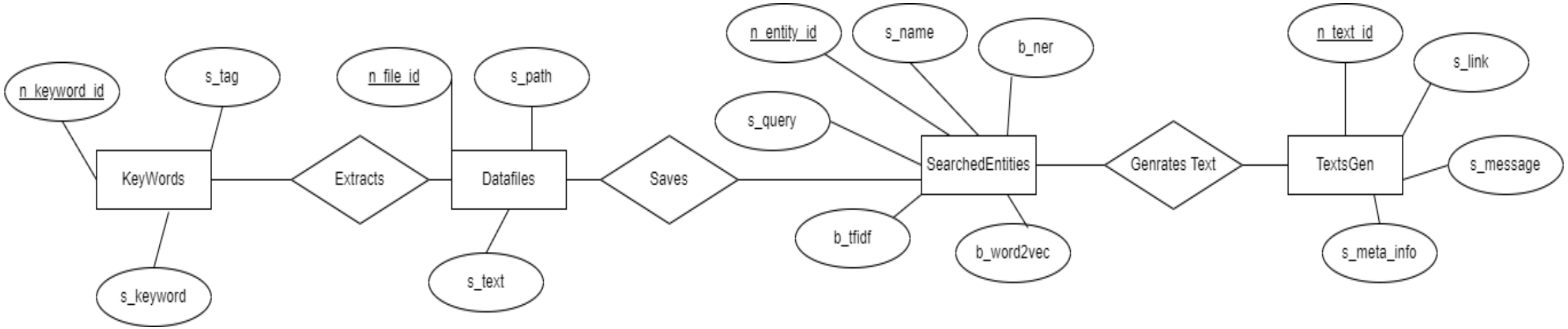
Zu speichern:

- Gefundene Keywords jeder Entity
- Durchsuchte PDFs mit OCR-Output
- Generierter Text

4. Umsetzung

Zu speichern:

- Gefundene Keywords jeder Entity
- Durchsuchte PDFs mit OCR-Output
- Generierter Text



4. Umsetzung

Ziele:

- Responsive Website
- Intuitiv
- Visuelle Darstellung der DB
- Visuelle Darstellung der generierten Texte

Umsetzung:

- Bootstrap
- Flask
- SQLAlchemy

5. Fazit

SeemsPhishy Workflow:

Mögliche Erweiterung:

- Verschiedene Textgeneration Templates
- Automatische Evaluierung der Keywords
- Erweiterte Data Extraction Pipeline
- Big Data Analysis → Mehr Daten Berücksichtigen

Vielen Dank für ihre Aufmerksamkeit