SEBA Lab WS2022/23
Felix Hoops, Munich, 18th of October 2022

Chair of Software Engineering for Business Information Systems (sebis)
Faculty of Informatics
Technische Universität München
www.matthes.in.tum.de
Language Interface for Semantic Search Assistance (LISSA)

Advisor / Product Owner: Phillip Schneider
Language Interface for Semantic Search Assistance (LISSA)

Outcome: Conversational interface that provides:
- ✓ Assistance for semantic search
- ✓ Intelligent dialog management
- ✓ Retrieval of structured and unstructured data

Technologies
- Flask
- Hugging Face
- haystack by deepset
- RDF
- neo4j
- RASA
- RASA Core
- RASA NLU
- Action Server
- Knowledge Base

LISSA Frontend

Hi, how can I help you?

Are there any papers on knowledge graphs in the latest ACL proceedings?

Yes, I found 8 papers from ACL '22 that could be relevant. Here are the two most-cited ones:

1. KG-FiD: Infusing Knowledge Graph in Fusion-in-Decoder for Open-Domain Question Answering (Yu et al.)
2. Sequence-to-Sequence Knowledge Graph Completion and Question Answering (Saxena et al.)

Show more information
Pitch: The goal of the project is to develop a conversational interface that supports semantic search for unstructured as well as structured data in scientific knowledge bases.

**Expected prior knowledge**
- Good knowledge of Python programming language
- Ability to manage and query different data structures
- Strong interest in NLP and at least basic NLP knowledge

**Basic functional requirements**
- Interactive navigation through data source by means of a simple, text-based dialog interface
- Information retrieval of structured as well as unstructured data records
- Intelligent NLU and dialog management for enabling information-providing conversations

**Basic non-functional requirements**
- Self-explanatory UI and engaging UX & dialog design
- Modular and reusable components
- Efficient query processing in realtime
- Scalable system architecture

**Bonus**
- Gain knowledge about state-of-the-art NLP techniques
- Learn how to engineer conversational interfaces with widely-used open-source tool stacks
- Get guidance and feedback from experienced industry professionals

**Contact**
philip.schneider@tum.de
WorKSumm – World Knowledge Summarization Skill for chatbot

Advisor / Product Owner: Anum Afzal
Outcome: A chatbot that supports:

- Understanding of user utterance.
- Retrieval of relevant data for the user prompt.
- Summarization of the world knowledge into concise text for the chatbot

Technologies:

- Hugging Face
- TensorFlow
- PyTorch
**Pitch:** The goal of the project is to create a world knowledge skill for a chatbot. It involves using Natural Language Processing algorithms to summarize a topic-related world knowledge into a concise response to be used by the chatbot.

<table>
<thead>
<tr>
<th>Expected prior knowledge</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Experience in Python Programming Language.</td>
</tr>
<tr>
<td>- Knowledge in Natural Language Processing (NLP) and Deep Learning (DL) Methodologies.</td>
</tr>
<tr>
<td>- Basic Knowledge in Hugging Face, PyTorch or TensorFlow</td>
</tr>
<tr>
<td>- Experience in Conversational AI and Rasa [optional]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Basic functional requirements:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Creation of a world knowledge skill for a chatbot.</td>
</tr>
<tr>
<td>- Analysis and Pre-processing of the dataset.</td>
</tr>
<tr>
<td>- Evaluation of state-of-the-art Abstractive Text Summarization models for large documents.</td>
</tr>
<tr>
<td>- Implementation of the best-performing Text Summarization model inside the world knowledge skill.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Basic non-functional requirements:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- The summarization skill is able to generate concise summarizes of a topic.</td>
</tr>
<tr>
<td>- Implementation of NLG evaluation metrics to access the quality of the summaries.</td>
</tr>
<tr>
<td>- Implementation of fallback mechanism for Intents</td>
</tr>
</tbody>
</table>

**Contact:**
anum.afzal@tum.de
Fact-Checking Tool

Advisor / Product Owner: Juraj Vladika
Claim: Coffee helps increase lifespan

People who drink up to eight cups of coffee per day may slightly lower their risk of early death compared to non-drinkers, according to a large new study. And it doesn’t appear to matter if the coffee is caffeinated or decaf, brewed or instant. The study analyzed data from about half a million Britons and found that the more coffee people drank, the lower their risk of dying during the 10-year study period. “People think of caffeine, but it’s likely that some of the most beneficial compounds are not the caffeine.” (...

source: harvard.edu

Claim: Koalas have gone extinct in 2019 after bushfires hit Australia.

Over the weekend, erroneous declarations that the animals have lost most of their habitat and are “functionally extinct” made the rounds in headlines and on social media. Koalas are considered vulnerable to extinction—just a step above endangered—and reports indicate that many of them have been found dead so far in fire-devastated zones. But, experts say, we are not looking at the death of a species—yet. “We’re not going to see koalas go extinct this fast,” says Chris Johnson, professor of wildlife conservation at the University of Tasmania. (...

source: nationalgeographic.com

Claim: Coconut oil can cure Alzheimer’s disease.

The damage caused by Alzheimer’s disrupts the brain’s ability to use its primary energy source, known as glucose. Ketones may provide an alternative energy source to the brain cells to moderate the damage caused by Alzheimer’s disease. The body produces ketones when it metabolizes coconut oil and similar substances. Unfortunately, there just isn’t any credible science to support this idea. It is impossible for us to know how well coconut oil does or does not work in Alzheimer’s disease because there have not been rigorous, large-scale research studies done. (...

source: alz.org
Pitch: The goal of the project is to develop an automated fact-checking tool that for an input claim finds evidence in the corpus of documents or websites and produces a verdict on its veracity, using integrated machine learning models.

Basic functional requirements:
- Creation of a graphical UI for input and navigation
- Visualization of found evidence and verdicts
- Preparation and processing of data
- Construction and configuration of machine learning models

Basic non-functional requirements:
- Realtime processing of predefined and new queries
- Efficient retrieval of documents from the database
- Modularity and reusability of components

Expected prior knowledge
- Knowledge of HTML, JavaScript, CSS
- Skills in Python, Flask / Django, Angular / React
- Integration with SQLite, Google Cloud / AWS

Desirable:
- Knowledge of ML models, PyTorch, Hugging Face
- Understanding of NLP methods & applications

Contact:
juraj.vladika@tum.de
Research Institution Knowledge Graph

Advisor / Product Owner: Tim Schopf
Outcome: App that supports:
✓ Linking of related scientific domains
✓ Visual knowledge exploration

Technologies:
- Flask
- Neo4j
- Django
Pitch: The goal of the project is to develop a web application which provides an explicit overview of the research fields within a research institution based on graph data.

Basic functional requirements:
- Support navigation of research concepts based on a provided ontology
- Support search of researchers and their topics

Expected prior knowledge
- Knowledge in React, Python and Flask
- Basic Knowledge in Neo4j beneficial

Basic non-functional requirements:
- Realtime processing of predefined queries
- Adaption of the result set in realtime
- Realtime client – server communication
- Design of modular components

Contact:
tim.schopf@tum.de
DP/PCE
Data Privacy – Privacy Compliance Ecosystem

Advisor / Product Owner: Alexandra Klymenko
Stephen Meisenbacher
Outcome:
- App that supports:
  - Educational UI
  - Interaction with learning content
  - Visual exploration
  - Decision support system
  - Discussion forum

Technologies:
- Flask
- React
- HTML
- CSS
- JS

In this Decision Tree, you receive recommendations based on the answers you provide. As a result, accuracy of recommendations is dependant on the their correctness. You can answer with:

- **YES**: You receive further questions regarding technologies that possess the inquired characteristic.
- **NO**: You receive further questions regarding technologies that do not possess the inquired characteristic.
- **SKIP**: You receive further questions regarding technologies that do and do not possess the characteristic. You should skip questions if you are not certain about the implication or description provided.

Do your Computations rely on real values you have collected previously?

- **YES**: Most value generation or computations processes directly run on the actual data. Using actual data sets increases accuracy and can produce more closely related to concrete values.
- **SKIP**: You should skip questions if you are not certain about the implication or description provided.
- **NO**: You receive further questions regarding technologies that do and do not possess the characteristic.
Outcome: App that supports:
- Educational UI
- Interaction with learning content
- Visual exploration
- Decision support system
- Discussion forum
**Pitch:** The goal of the project is to extend a learning platform for Privacy-Enhancing Technologies with a focus on interactively showcasing the data privacy compliance process.

**Basic functional requirements:**
- Integration of an existing decision support system
- Linking of the decision support system to existing learning material on PETs
- Creation of interactive privacy compliance structure exploration tool
- Addition of a discussion forum for registered users
- Further e-learning features (e.g., gamification)
- UI design improvements

**Expected prior knowledge:**
**Required:**
- Knowledge in HTML, CSS, JavaScript, Material UI, React, Python, Flask, MongoDB

**Desirable:**
- First experience or high interest in data privacy (compliance) and Privacy-Enhancing Technologies

**Basic non-functional requirements:**
- Intuitive user interface
- Code quality
- Modular and extendable design

**Contact:**
alexandra.klymenko@tum.de
stephen.meisenbacher@tum.de
DP²NLP
Data Privacy – Differential Privacy in Natural Language Processing

Advisor / Product Owner: Stephen Meisenbacher
Alexandra Klymenko
DP²NLP – Explore Word-Level Text Privatization

**Goal:** to develop an educational user interface that allows users to explore the application of word-level Differential Privacy to textual data, in an interactive and real-time manner.

**Outcome:** App that supports:
- Interactive NLP Dashboard
- Real-time calculations
- Complex Data Visualization
- Educational UI

**Technologies:** Flask, Plotly, Dash, HTML, CSS, JavaScript

---

**Framework**

**Hyperparameters**

**EMBEDDING SPACE VISUALIZATION**

**MECHANISM CALCULATIONS**

<table>
<thead>
<tr>
<th>D'</th>
<th>$\exp\left(\frac{eq(D, r)}{2\Delta q}\right)$</th>
<th>Pr[D']</th>
<th>q(heart, D')</th>
</tr>
</thead>
<tbody>
<tr>
<td>liver</td>
<td>16.602</td>
<td>0.367</td>
<td>0.590</td>
</tr>
<tr>
<td>lung</td>
<td>9.647</td>
<td>0.213</td>
<td>0.476</td>
</tr>
<tr>
<td>tissue</td>
<td>6.560</td>
<td>0.145</td>
<td>0.395</td>
</tr>
<tr>
<td>diabetes</td>
<td>6.315</td>
<td>0.140</td>
<td>0.387</td>
</tr>
<tr>
<td>cancer</td>
<td>6.108</td>
<td>0.135</td>
<td>0.380</td>
</tr>
</tbody>
</table>

---

**ORIGINAL TEXT**

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. *heart* Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

**PRIVATE TEXT**

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. *liver* Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
**Pitch:** The goal of the project is to develop an educational user interface that allows users to explore the application of word-level Differential Privacy to textual data, in an interactive and real-time manner.

<table>
<thead>
<tr>
<th>Expected prior knowledge:</th>
</tr>
</thead>
<tbody>
<tr>
<td>▪ Knowledge in Flask, React, Javascript, HTML, CSS</td>
</tr>
<tr>
<td>▪ Strong programming skills in Python</td>
</tr>
<tr>
<td>▪ Basic knowledge in Natural Language Processing</td>
</tr>
<tr>
<td>▪ Solid knowledge in Probability &amp; Statistics</td>
</tr>
<tr>
<td>▪ Genuine interest in (data) privacy and Privacy-Enhancing Technologies!</td>
</tr>
<tr>
<td>▪ Experience in JavaScript / Python visualization libraries like D3, Chart, Plotly, etc.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Basic functional requirements:</th>
</tr>
</thead>
<tbody>
<tr>
<td>▪ Creation of an interactive text privatization playground</td>
</tr>
<tr>
<td>▪ Displaying of perturbation results, both graphically and mathematically</td>
</tr>
<tr>
<td>▪ Exhibition of non-privatized vs. privatized text</td>
</tr>
<tr>
<td>▪ Providing relevant stats and user notifications</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Basic non-functional requirements:</th>
</tr>
</thead>
<tbody>
<tr>
<td>▪ Real-time execution of Differential Privacy</td>
</tr>
<tr>
<td>▪ Intuitive visualization of the perturbation process</td>
</tr>
<tr>
<td>▪ Adaptation of calculations to user-given parameters</td>
</tr>
<tr>
<td>▪ Design of custom components to facilitate the UI</td>
</tr>
<tr>
<td>▪ Code readability and quality</td>
</tr>
</tbody>
</table>

**Contact:**

- [stephen.meisenbacher@tum.de](mailto:stephen.meisenbacher@tum.de)
- [alexandra.klymenko@tum.de](mailto:alexandra.klymenko@tum.de)
Governance Evaluation for Algorand

Advisor / Product Owner: Burak Öz
Christian Ziegler
SUPPRA-Governance – Governance evaluation for Algorand

Governance Period Timeline

Governance works in cycles. Each cycle lasts three months and contains a sign up phase, a voting phase and a rewards phase.

Outcome: App that supports:
✓ Governance Visualization
✓ Quick Pagespeeds
✓ Real-time updates
✓ Data aggregation
Pitch: The goal of the project is to develop a governance evaluation platform and dashboard for the cryptocurrency Algorand with ongoing data-collection.

Expected prior knowledge
- Knowledge in tailwindcss, Next.js, NodeJS
- Basic Knowledge working with REST APIs, GraphQL, RPCs.
- Experience in JavaScript visualization libraries d3
- Basic Knowledge of Socket.io or similar realtime client – server communication frameworks

Basic functional requirements:
- Fetch Governance data from the Algorand Blockchain
- Aggregate Governance Data and do basic analysis
- Visualize the aggregated Governance Data
- Design an ongoing data-collection

Basic non-functional requirements:
- Quick loading times and very good Pagespeed scores
- Realtime client – server communication
- Design of modular components
- Caching of results in the server

Contact:
burak.oez@tum.de
christian.ziegler@tum.de
Group Ordering

Advisor:  Felix Hoops
Product Owner:  Isa Usmanov
E-commerce and delivery solution for SMEs

Step 1
Choose a unique link for your shop
https://company.droov.io

Step 2
Drag-and-drop assemble and design your store (no code)

Step 3
Configure logistics to meet your business needs
- Define delivery area
- Set operating days and time
- Connect payment methods
- Organise & manage couriers
Goals – Group ordering & business analytics dashboard
Pitch: droov allows small businesses to effortlessly create their online store and operate a delivery business. The two main goals of this project are 1) Extending current solution with a possibility to offer a group ordering functionality. For example, if co-workers in the office or a group of friends in Englischer Garten want to order together from a single restaurant. Group orders are economically and ecologically beneficial for all of the parties involved. 2) Developing a business analytics dashboard to display business performance metrics, among others, number of daily/weekly orders, revenues etc.

Basic functional requirements:
- As group organizer, initiate a group order, generate invite link & QR-code. Joining group via link; checkout
- Paying and submitting batched order and creating a group order status tracking view.
- Configuring group ordering details in the admin view
- Analytics dashboard with business performance metrics

Basic non-functional requirements:
- Intuitive and responsive user interface (Always eliminate extra clicks where possible - be creative)
- Follow droov’s styling pattern
- Collaborate and engage with founder team

Expected prior knowledge
- Prior knowledge in web application & front-end development React.js / JavaScript
- Basic experience with styling CSS libraries
- Prior knowledge in backend development (Node.js or Firebase + MongoDB or Firestore DB)
- Basic knowledge in version controlling tools: Git / Gitlab
- Extra points for experience with Payment processing solutions (not mandatory)

Contact:
Isa Usmanov - isa@droov.io
SHAREVAX – Get and share the vaccines people need

Advisor: Peter Kuhn
Product Owner: Monja Puggel
Martin Gail
SHAREVAX – Get and share the vaccines people need

Overview about vaccines in stock

Outcome: App that supports:
✓ Calculation Algorithms
✓ Real-time Result Change
✓ Complex Event Processing
✓ Multi-level Data visualization

Technologies:
- TypeScript
- Angular
- HTML
- CSS
- Node.js
- Spring Boot
- MongoDB
Pitch: The goal of the project is to develop a web application for simulating the distribution of vaccines based on an algorithm that calculates optimal shipping routes and delivery.

Basic functional requirements:
- Simulating the usage of shipment routes
- Find optimal distribution of vaccines
- Visualization of vaccine amount and shipment routes
- Visualization of dashboards

Basic non-functional requirements:
- Processing of user input and adaption of the result in real-time
- Real-time client – server communication
- Design of modular components

Expected prior knowledge
- Knowledge in frontend technologies like Angular or React, HTML, CSS
- Knowledge in backend technologies like mongoDB, Spring Boot, NodeJS
- Basic knowledge in developing calculation algorithms

Contact:
- martin.gail@capgemini.com
- monja.puggel@capgemini.com
- p.kuhn@tum.de
CLIQ – A Web3 protocol to tokenize reciprocity in a social network

Advisor: Sascha Nägele
Product Owner: Zied Bahrouni
Ehsan Olyaee
CLIQ – A Web3 protocol to tokenize reciprocity in a social network

Outcome: A protocol that:
✓ Tokenizes reciprocity in the social network app Cliq
✓ Incentivizes users to generate value for each other
✓ is resilient against fraud
✓ Offers a different engagement metric in social networks (as opposed to time per user)

Technologies:
Pitch: Cliq (in beta) is a social network that enables users to create small closed cliques to exchange favors and organize collective action. The goal of this project is to create a Web3-based protocol that tokenizes reciprocity. The protocol would incentivize users to help others and with that create reciprocity within the group.

Basic functional requirements:
- Tokens should be earned by fulfilling favor requests, taking part in collective actions or nominating new members.
- The sum of tokens a user has therefore corresponds to their helpfulness within a group.
- Moreover, the tokens should be linked to privileges or financial incentives to further incentivize users to create value for each other.

Basic non-functional requirements:
- Create different concepts of the reciprocity token and test them with the Dev team and the social science team.
- Run experiments with Dev team and social science team in controlled groups of pilot users.

Expected prior knowledge
- General knowledge about web3 protocols and blockchain (bonus: in social networks)
- Ethereum protocol
- (Optional) Python, Django, JavaScript, React
- Basic knowledge in DevOps

Contact:
zied.Bahrouni@motius.de
ehsan.olyaee@motius.de
sascha.naegele@tum.de
Know your Money! – Budget Calculator with PSD2

Advisor: Pascal Philipp
Product Owner: Francisco De las Casas Young
Krassimir Ovcharov
Felipe Wieman

interhyp
Know your Money! – Budget and Estate Platform with PSD2

**Outcome – Web-App that supports:**
- Integration of Estate, Financing and Bank APIs
- Calculations of the customers financial possibilities
- Categorization and optimization of income and expenses
- Multi-level data visualization of complex financial data
- Personalized mortgage calculations
- Personalized estate recommendations

**Technologies:**

- Integration of customers bank data
- Integration of financing data
- Integration of estate data
Pitch: The goal of the project is to develop a budget and estate web platform using different APIs to compute the financing possibilities of our customers to provide recommendations for estates and mortgages.

Basic functional requirements:
- Implementation of several forms and views
- Visualization of data and calculations
- Integration of a PSD2 (Payment Service Directive 2), Interhyp and Estate APIs
- Categorization and Optimization of income/expenses
- Personalized recommendations for estates and mortgages

Expected prior knowledge
- Knowledge in Frontend Technologies
  - HTML, CSS, JavaScript, TypeScript, React
- Knowledge in Backend Technologies
  - Java, Node, Kotlin, Spring Boot
- Bonus: Knowledge of PSD2 conform Bank APIs

FYI: If you want to choose another technology that is not listed here, just let us know! :)

Basic non-functional requirements:
- Intuitive, performant and responsive user interface (mobile first)
- Design of modular and reusable components
- Security (2FA, Captcha, Access Token)

Contact:
francisco.delascasasyoung@interhyp.de
krassimir.ovcharov@interhyp.de
felipe.wieman@interhyp.de
pascal.philipp@tum.de
Managing cloud developer workspaces

Advisor: Tri Huynh
Product Owner: Jonas Helming
Managing cloud developer workspaces

Outcome:
- App that supports:
  - Managing Dev Workspaces
  - Monitoring Deployments
  - Resource/Cost Analysis
  - Optimizing the cluster

Technologies:
- Browser-based IDE
- Cluster of developer workspaces (Kubernetes)
Pitch: The goal of the project is to develop a web application for managing a cluster hosting developer workspaces.

Basic functional requirements:
- Configure parameters, such as allowed resource consumption per user
- Managing user workspaces
- Monitoring deployed service
- Statistics about usage, performance and cost
- Manage cluster optimizations, e.g. “prewarming”

Expected prior knowledge
- Knowledge in HTML, CSS, TypeScript and React
- Basic Knowledge in Docker and Kubernetes
- Standard “toolbox” including Git and VS Code

Basic non-functional requirements:
- All code will be contributed under an Open Source license (EPL)
- Reproducible build and set-up

Contact:
jhelming@eclipsesource.com
tri.huynh@tum.de
Felix Hoops (M.Sc.)
Research Assistant

Technische Universität München
Faculty of Informatics
Chair of Software Engineering for Business Information Systems

Boltzmannstraße 3
85748 Garching bei München

Tel +49.89.289.17114
Fax +49.89.289.17136

felix.hoops@tum.de
www.matthes.in.tum.de