NL search for semantic web

Lukas Kleine Büning
Pichaya Kanjanapisith
Yuchun Chen
Venkat

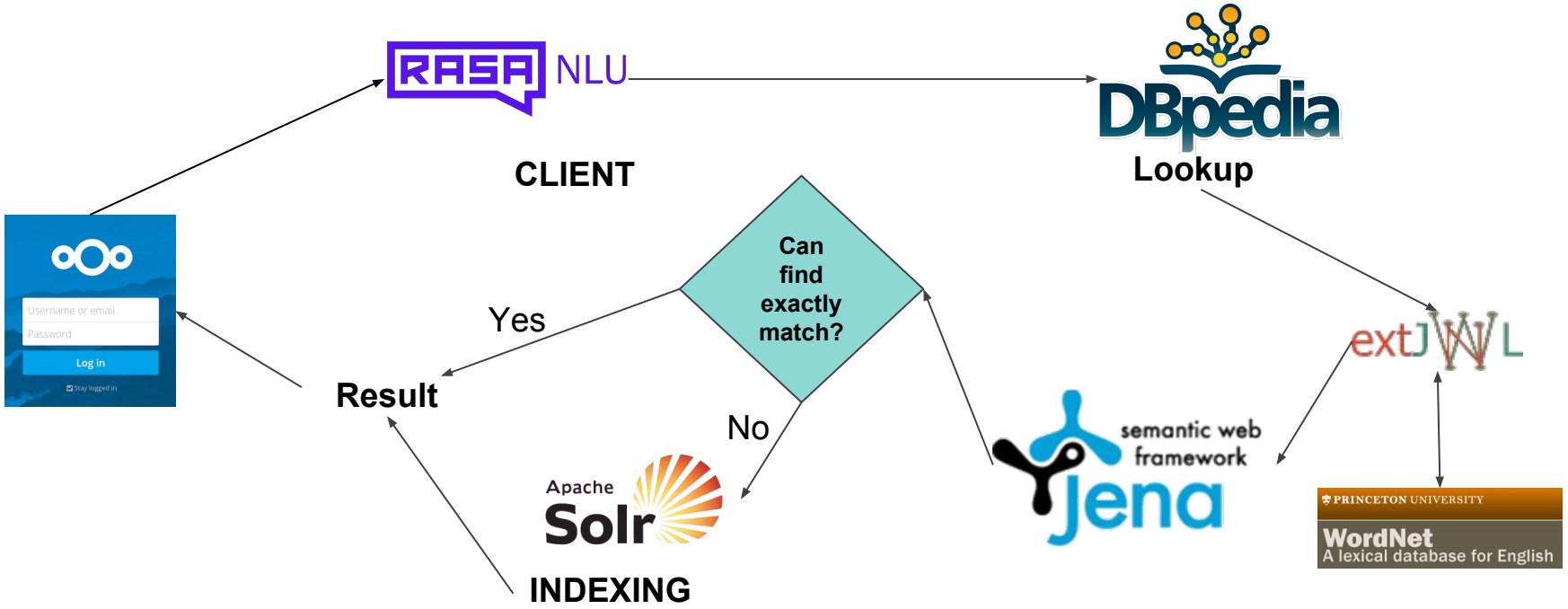
Agenda

- Motivation & Goals
- Component
- Demo
- Outcome
- Future Work
- Responsibility

Motivation & Goals

- **Motivation:**
  - Searching in semantic sources requires special knowledge (SPARQL, ...)
  - General users cannot gain any benefit from such data sources
  - Direct search on semantic data host can timeout and don't scale as number of user queries grow.
- **Goals**:
  - Make search easy through identifying entities with NLU.
  - Lookups on Dbpedia
  - Intermediate index for failover.

Components : Application flow



Components

RASA for NLU processing

- Uses a linear **S**upport **V**ector **M**achine
- NLU by own definition ➡ own training data

  What is the capital of Germany?

Components

DBpedia Lookup : Subject measurement

- Web service that can be used to look up DBpedia URIs by related keywords.
- The results are ranked by the number of inlinks pointing.
- For example,
    - Subject input is “**USA**” will output as **United_States**
    - Subject input is “**Berl**” will output as **Berlin**.

Components

WordNet - Dictionary dataset for predicate

- English similarity dictionary linked by for synonym and generality of the word call synset.
- Synsets are interlinked by means of conceptual-semantic and lexical relations.
- For example:
    - {furniture, piece_of_furniture} the similarity will be increase to specific ones like {bed}.

.

Components

extJWNL - Predicate measurement

- Java API for creating, reading and updating dictionaries over WordNet.
- The key feature of this framework is easy to extent the dictionary dataset and possible to link it to the other data source.
- This procedure will receive predicate from RASA trying to find similarity word from the dictionary and output all possible word.
- For example:
    - It is possible to add dateOfBirth as the child of date of birth entity so when user input “What is the **date of birth** of Donald Trump?” it will be output **dateOfBirth** as one of the predicate.

Components

Apache Jena - SPAQL property query

- Open source framework for OWL and Semantic web.
- Use for query all property by the given Subject.
- The backend will find the property that match exactly with predicate then output as result.
- Example for all process in Lookup:
    - If the subject is **Berlin** and predicate is **capital**.
    - DBpedia searching for page Berlin correct format and exist or not.
    - Then, extJWNL will find all possibly words in dictionary.
    - Then Jena will search for page Berlin on DBperia and find all properties
    - If it can find the property match with any word in dictionary. It will return the results to users.
    - If no, pass new subject and all word in predicate combine with sentence and send it to Indexing procedure.

Components

Apache Solr - Indexing framework

- Solr is a standalone enterprise search server with a REST-like API.
- Solr index created once, can be used for fast look up of relavant documents.
- Solr framework also offers hit highlighting,spellcheck, auto-suggest features on the built index.

Index building

Search engines operate on pre-built “**inverted index**”.

- simplified model:
  - **corpus** is **represented as** *document x term* **matrix**
  - a cell *m,n* is 1 if document *m* contains term *n* and 0 otherwise

$$A = \begin{array}{c|ccc} & \textit{bike} & \textit{harley} & \textit{berlin} \\ \textit{doc1} & 1 & 0 & 0 \\ \textit{doc2} & 1 & 1 & 0 \\ \textit{doc3} & 0 & 0 & 1 \end{array}$$

- **queries** „harley“ and „harley bike“ **are just vectors in the term space** (analogous to documents)

$$q_1 = \begin{array}{ccc} \textit{bike} & \textit{harley} & \textit{berlin} \\ 0 & 1 & 0 \end{array} \qquad q_2 = \begin{array}{ccc} \textit{bike} & \textit{harley} & \textit{berlin} \\ 1 & 1 & 0 \end{array}$$

Slide from: AIM3, S.Schelter , DIMA

Outcome

- User could search with natural language query or chose from specific list of queries.
- Even if the input query does not match with training data, index will give some recommendations.
- Gain knowledge about semantic search, natural language processing and page indexing.

Future Work

- The extension
  - Front-ends
    - Pagination
    - Extend questions list based on NLU training datasets
  - RASA NLU
    - Increase training datasets
  - ExtJWNL dictionary
    - Increase dictionary coverage by extending data sources may be dbpedia
  - Solr indexing
    - Increase indexing pages
    - Hit highlight,Auto-suggest and spell checks

Demo

Responsibilities

- Lukas Kleine Büning
  - RASA NLU, Python Client
- Pichaya Kanjanapisith
  - Spring boots Server, DBpedia lookup, extJWNL , Jena Query
- Yuchun Chen
  - Webinterface , Python Client, Solr Indexing
- Venkat
  - Solr Indexing

Thank you!

Q&A

Repository :
https://gitlab.tubit.tu-berlin.de/pkanjan37/SW-LD_NLP_project

Reference

https://discuss.elastic.co/t/semantic-search-engine-on-the-top-of-es-any-suggestions-comments/41527

http://lucene.apache.org/solr/4_6_1/

http://mudassirshahzad.com/wp-content/uploads/2017/02/spring-boot.png

https://pbs.twimg.com/profile_images/578479910366269440/quS6q6Yu.png