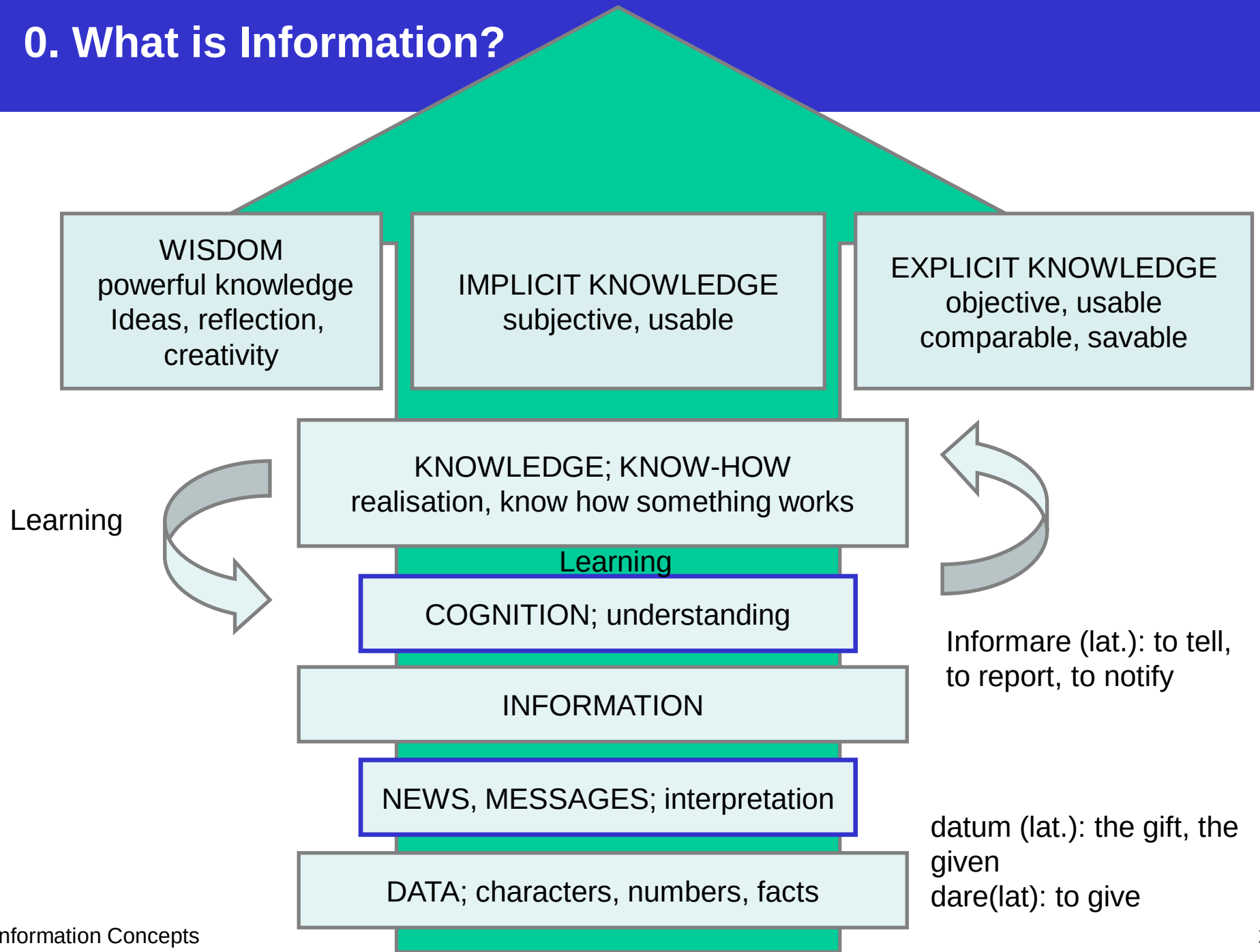
Mario Kubek
FernUniversität in Hagen, Germany
Tel.: +49 2331 987 4413
E-Mail: mario.kubek@fernuni-hagen.de

The Hagen NLPToolbox (March 2021 Edition with NLP Intro)

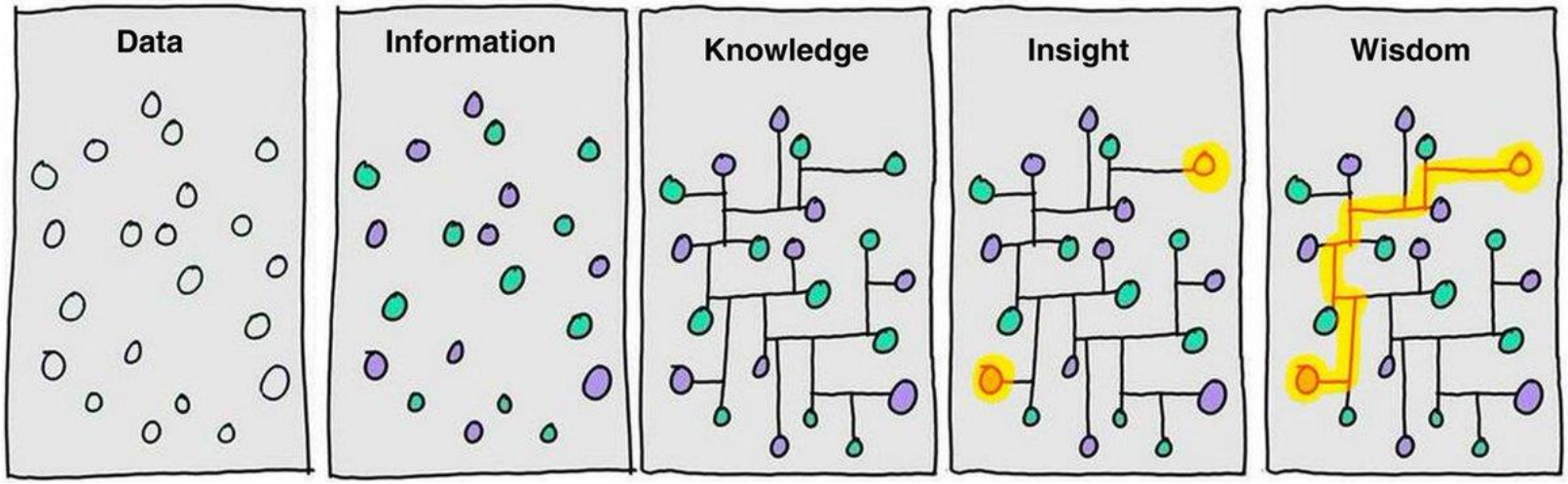
https://www.mario-kubek.de/lectures/The_Hagen_NLPToolbox_NLIR2021.pdf

1. **What is the Hagen NLPToolbox?**
2. **Pros and Cons**
3. **Practical Demonstration**
   - **Installation and Setup**
   - **Running in Different Modes**
   - **Graph Analysis in Neo4j**
4. **Recommended Alternatives**

0. What is Information?



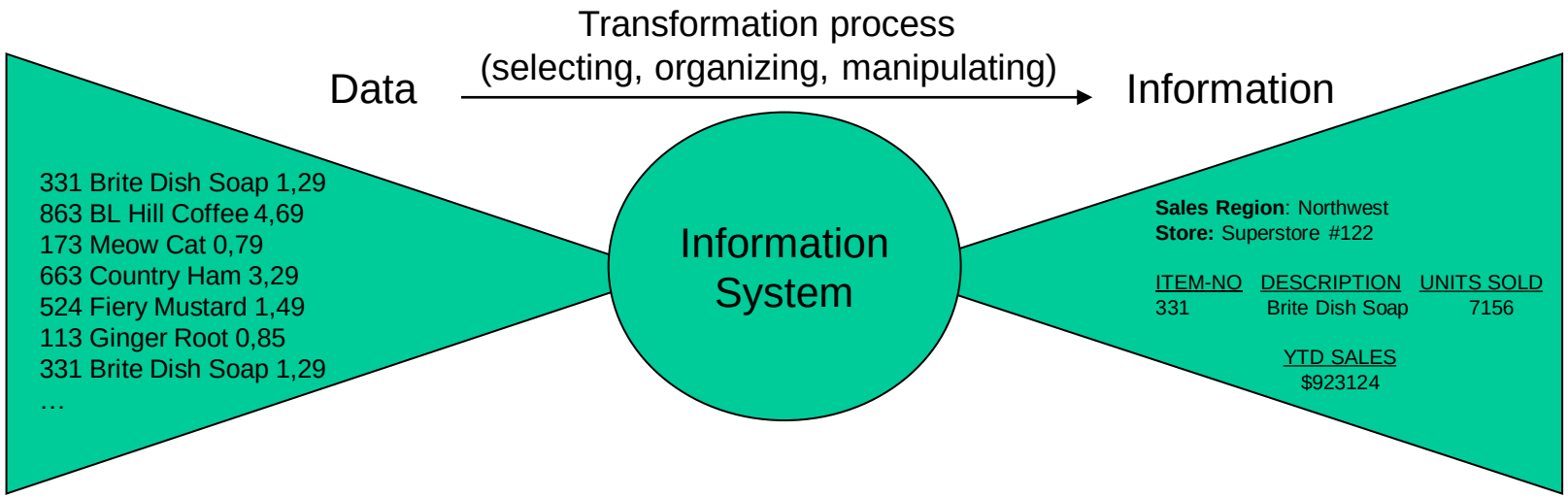
0. What is Information?



Source: http://www.verveiq.com/news/2016/12/5/creating-a-data-driven-culture

0. Definition of Text Mining (TM)

- Text Mining: "Process of deriving high-quality <u>information</u> from text" (Feldman & Sanger 2006)
  - Transformation of <u>data</u> (raw facts) into <u>information</u> (message that can be interpreted and understood by human beings) must occur. Also: information is the basis for <u>knowledge</u> (application of <u>information</u>).



0. Challenges in Natural Language Processing (NLP)

- But: Text is usually unstructured!
  - Keywords and basic concepts are unknown.
  - Their dependencies and relationships likewise.

- In Contrast to (Relational) Databases:
  - Data is structured according to a given schema.
  - High development costs!

- In order to <u>extract information</u>, text must be structured!
  - <u>Textual data must be preprocessed</u> and transformed such that it is turned into useable input (e.g. word vectors) for Text Mining methods.

0. Why is Text Difficult?

non-standard English

Great job @justinbieber! Were SOO PROUD of what youve accomplished! U taught us 2 #neversaynever & you yourself should never give up either♥

segmentation issues

the New York-New Haven Railroad
the New York-New Haven Railroad

idioms

dark horse
get cold feet
lose face
throw in the towel

neologisms

unfriend
retweet
bromance

world knowledge

Mary and Sue are sisters.
Mary and Sue are mothers.

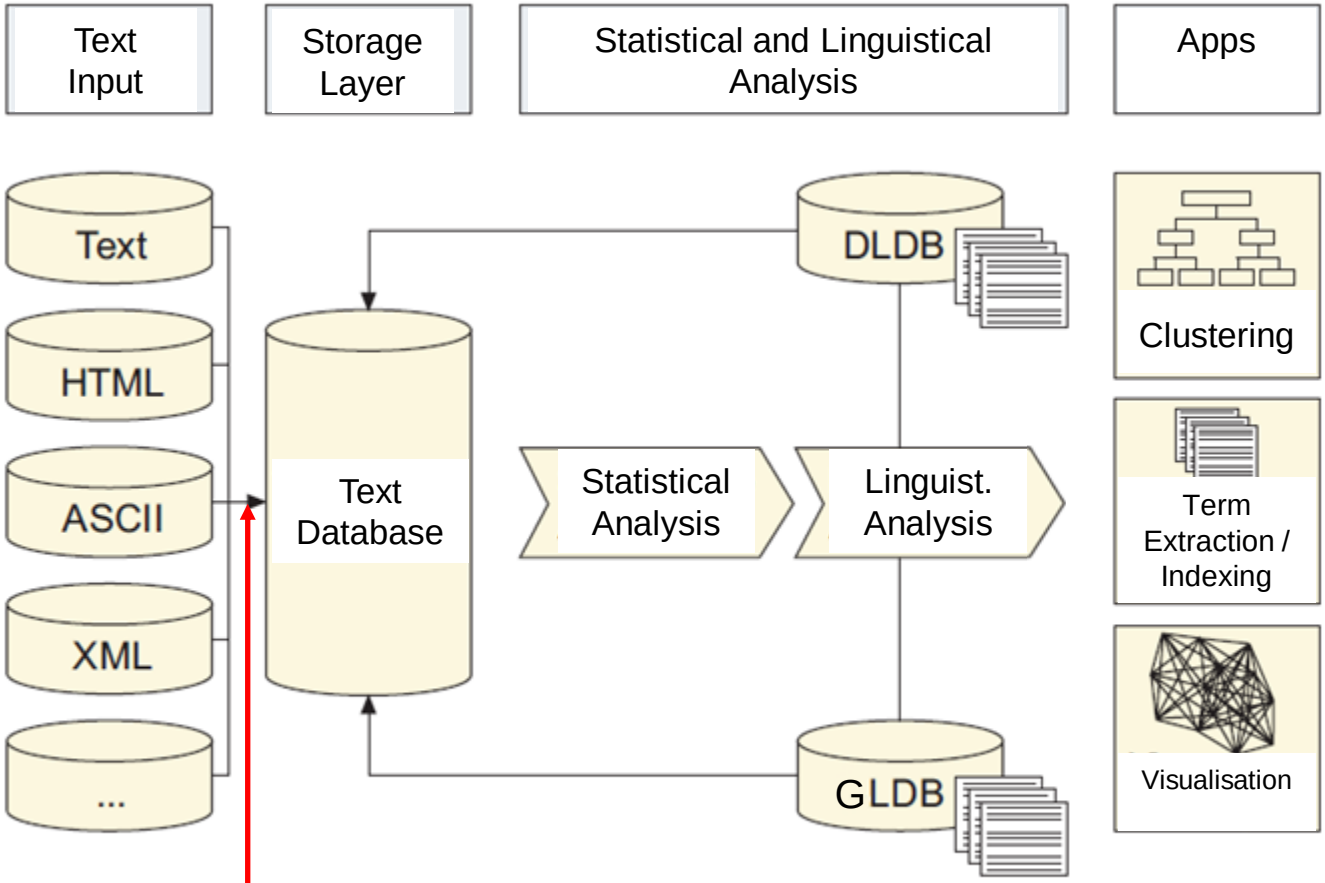
tricky entity names

Where is *A Bug's Life* playing ...
Let It Be was recorded ...
... a mutation on the *for* gene ...

Source: CS124 Stanford

- Languages: compound splitting in German:
  - Eierschalen|sollbruchstellen|verursacher
  - Baum|schulen|weg

0. General Text Analysis Layers



Preprocessing is carried out here!

Legend:
DLDB = Domain-specific linguistical database
GLDB = General linguistical database

(Source: Heyer et al., 2006)

0. Basic and Advanced Tasks in NLP

- Basic Tasks in Natural Language Processing:
  - Language detection and sentence / word segmentation
  - Part-Of-Speech tagging (nouns, verbs, adjectives, adverbs, card.) e.g. using Hidden Markov Models
  - Baseform reduction (e.g. houses->house)
  - Removal of stop words (and, the, of…) and other items
  - Term frequency and word length analysis
  - Extraction of keywords in text corpora by TF-IDF and difference analysis using well-balanced reference corpora
- Advanced Tasks related to Text Mining:
  - Clustering terms and documents
  - Classification of documents and Sentiment analysis

0. Some Common Part-of-Speech Tags

- Sample EN POS-tags from the Penn Treebank:
  https://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html

NN	Noun, singular or mass
DT	Determiner
VB	Verb, base form
VBD	Verb, past tense
VBZ	Verb, third person singular present
IN	Preposition or subordinating conjunction
NNP	Proper noun, singular
JJ	Adjective

1. What is the Hagen NLPToolbox?

- Java-based set of classes and methods for **local analysis** of German and English texts and corpora
- Provides a **full text analysis pipeline** (format conversion, language detection, sentence and word segmentation, POS-tagging, baseform reduction, stopword removal, data cleaning)
- Focus: graph-based **keyword/centroid extraction** based on the **analysis of co-occurrence graphs**
- Uses **Neo4j Embedded graph database** 4.1.3 for storing reference co-occ. graphs (of text corpora)

2. Pros and Cons

- The Good:
  - Easy setup, runs out-of-the-box (in IDE Eclipse), stable
  - Simple (no threads) and easily extendable pipeline
  - Many algorithms for graph-based text analysis included (ext. PageRank, ext. HITS, Centroid calculation, Evolving Centroids, Query Expansion by Spreading Activation)
  - Neo4j Embedded 4.1.3 already included (as library)
- The Bad:
  - Experimental and sometimes slow
  - Often "quick-and-dirty" code!!! (You have been warned!)
  - Code not well documented and commented (if at all)

Downloading Hagen NLPToolbox (Eclipse-Project, 468 MB)

https://www.mario-kubek.de/projects/Hagen_NLPToolbox_March2021.7z

3. Practical Demonstration (Part 1)

- Program Structure
- Installation (in Eclipse using project import) / Setup
- Work Modes: single text analysis (mode 0) and graph DB generation (mode 1) in main()-method
  - Important classes: **TextProcessing.java** (main) and **Cooccs.java**
  - Co-occurrence graph creation and update using Neo4j embedded
  - Keyword extraction (nouns and names), centroid determination
- Result Output (CSV Files) and Interpretation
- Graph Database Export

3. Practical Demonstration (Part 1)

- Program Structure (after unpacking)

SSD-Daten (D:) › Ressourcen › Hagen_NLPToolbox_March2021 ›

Name	Änderungsdatum	Typ	Größe
.settings	04.03.2021 13:56	Dateiordner	
bin	04.03.2021 13:56	Dateiordner	
config	04.03.2021 13:56	Dateiordner	
cooccsdatabase	04.03.2021 13:56	Dateiordner	
corpora	04.03.2021 13:57	Dateiordner	
data	04.03.2021 13:56	Dateiordner	
download	04.03.2021 13:56	Dateiordner	
lib	04.03.2021 13:56	Dateiordner	
logs	11.04.2015 04:06	Dateiordner	
Neo4jLibs	04.03.2021 13:56	Dateiordner	
Neo4jLibs322	04.03.2021 13:56	Dateiordner	
output	17.06.2015 04:34	Dateiordner	
resources	04.03.2021 13:56	Dateiordner	
src	04.03.2021 13:56	Dateiordner	
.classpath	01.11.2020 12:56	CLASSPATH-Datei	13 KB
.project	15.03.2015 07:25	PROJECT-Datei	1 KB
log4j.out	14.05.2019 05:54	OUT-Datei	0 KB

3. Practical Demonstration (Part 1)

- Data folder, input, output, sentence files (satzfiles):

SSD-Daten (D:) › Ressourcen › Hagen_NLPToolbox_March2021 › data ›

Name	Änderungsdatum	Typ
indexes	25.12.2017 03:28	Dateiordner
input	04.03.2021 13:56	Dateiordner
output	04.03.2021 13:56	Dateiordner
satzfiles	04.03.2021 13:56	Dateiordner
termvectors	30.07.2017 14:03	Dateiordner

- Data input folder (single texts and corpora):

SSD-Daten (D:) › Ressourcen › Hagen_NLPToolbox_March2021 › data › input

Name	Änderungsdatum	Typ	Größe
txt	06.03.2021 11:45	Dateiordner	
Secure by default - Kopie.html	04.03.2021 22:12	Firefox HTML Doc...	11 KB
Secure by default.html	04.03.2021 22:12	Firefox HTML Doc...	11 KB

- Sentence files:

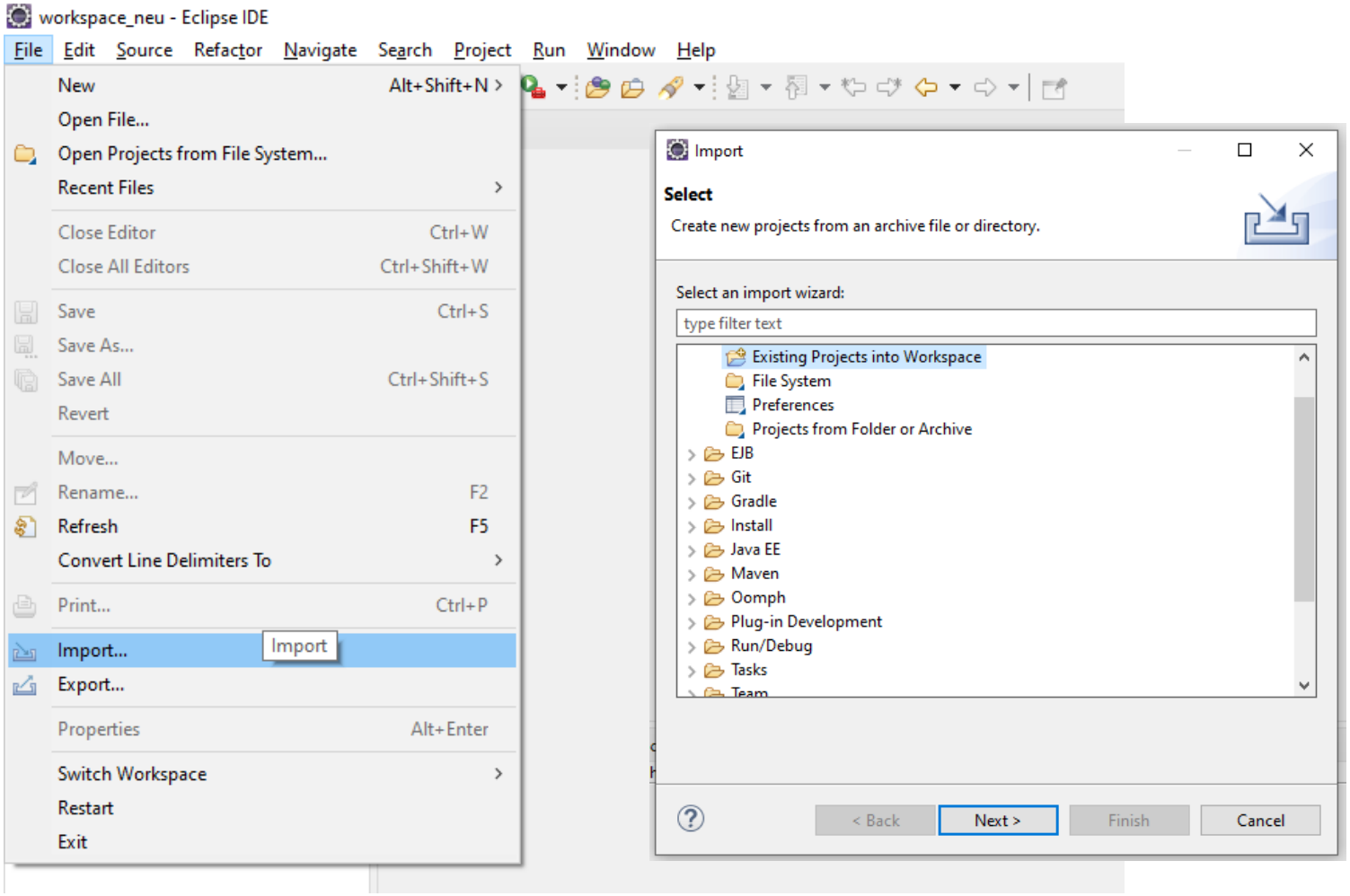
  Each line contains one sentence.

SSD-Daten (D:) › Ressourcen › Hagen_NLPToolbox_March2021 › data › satzfiles

Name	Änderungsdatum	Typ	Größe
Secure by default - Kopie.html.txt.s	06.03.2021 11:50	Assembler Source	4 KB
Secure by default.html.txt.s	06.03.2021 11:50	Assembler Source	4 KB

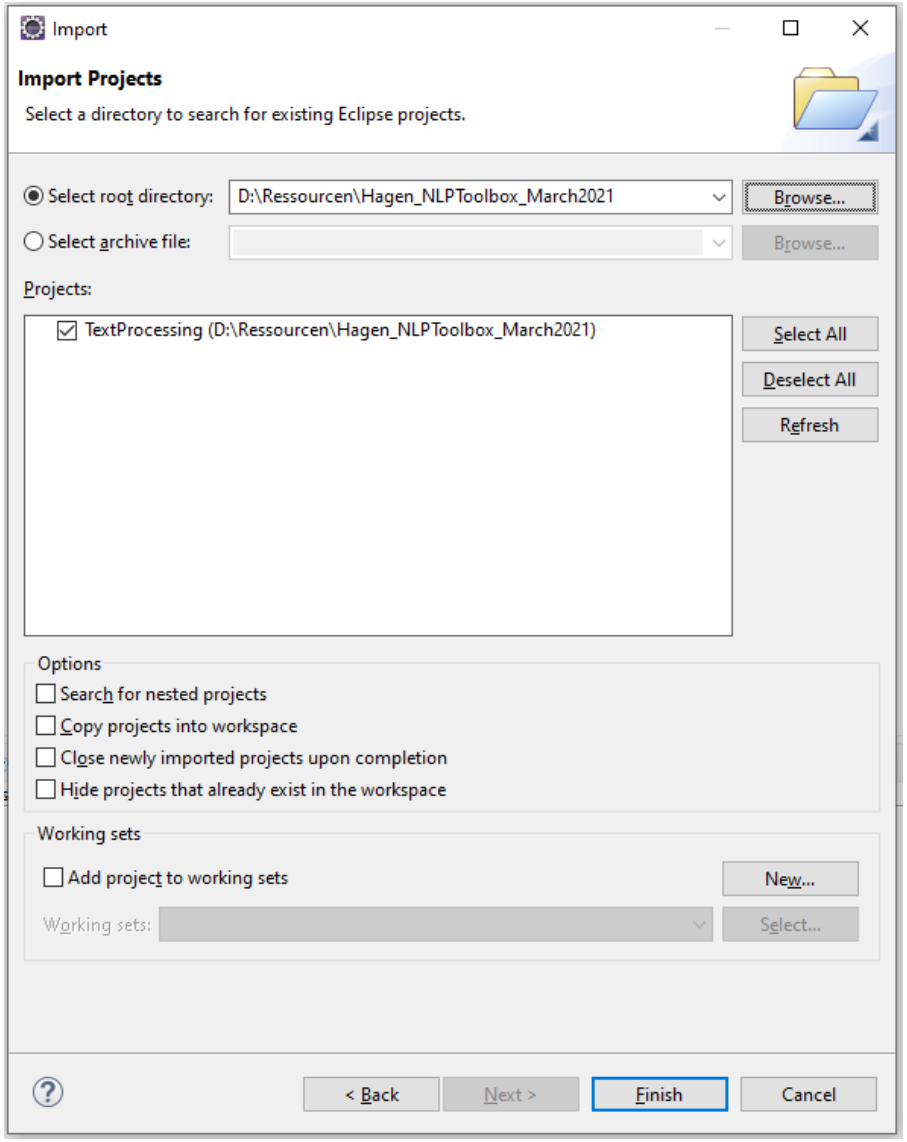
3. Practical Demonstration (Part 1)

- Importing the project in Eclipse (after unpacking)



3. Practical Demonstration (Part 1)

- Importing the project in Eclipse (after unpacking)



3. Practical Demonstration (Part 1)

workspace_neu - TextProcessing/src/core/TextProcessing.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

TextProcessing.java | Cooccs.java

```
1013 public static void main(String[] args) {
1014
1015     TextProcessing textprocessing = new TextProcessing("data"); // base
1016
1017     // Normally, run mode 1 first and then mode 0
1018     // Otherwise, centroid calculations will fail
1019     // (as no underlying corpus graph database has been created yet)
1020
1021     int mode = 1; // mode=0 analyse single files, mode=1 analyse corpus
1022                   // (to create local co-occurrence graph db)
1023
1024     if (mode == 0) { // Analyse single documents in data/input using
1025                      // curr. Neo4j graph database created using mode 1
1026
1027         long start = System.currentTimeMillis();
1028
1029         textprocessing.preprocessing(mode);
1030         textprocessing.analysis();
1031
1032         // Extra: Calculate document similarity if needed
1033         System.out.println("Similarity between document 1 and 2 is: "
1034                                 + textprocessing.docsim(0, 1));
1035
1036         long end = System.currentTimeMillis();
1037
1038         System.out.println("File processing took " + (end - start) / 10
1039
1040     } else if (mode == 1) { // Create Neo4j graph database only from al
```

Pac... | Typ...

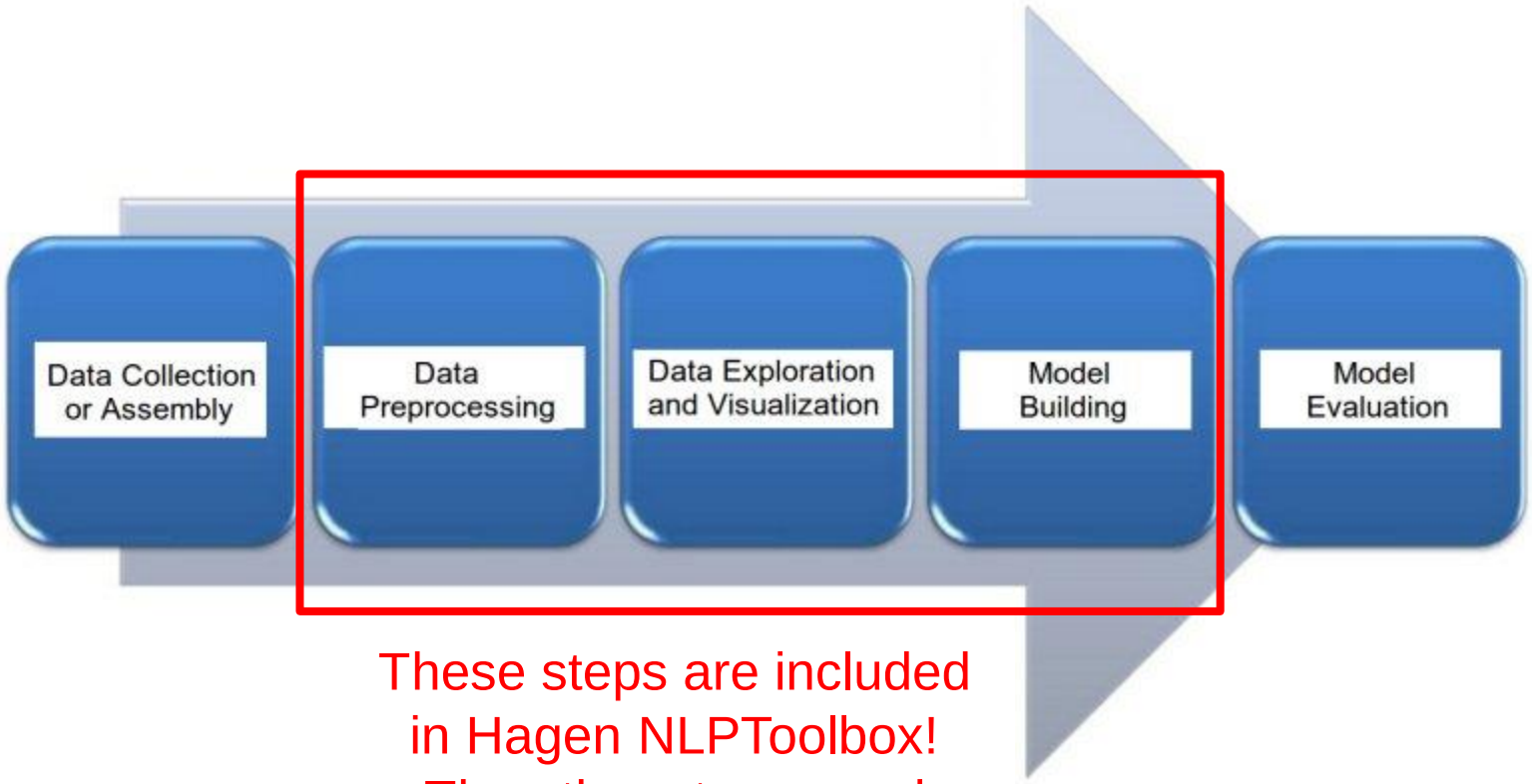
- corecrawler
- Pawawap
- pongGame
- SeleniumTest
- TextProcessing
  - src
    - (default package)
    - core
      - Cooccs.java
      - Labels.java
      - RelationshipTypes.jav
      - TextProcessing.java
    - de
    - de.texttech
    - de.texttech.cc
    - de.uni_leipzig
    - de.uni_leipzig.asv
    - de.uni_leipzig.asv.toolbo
    - de.uni_leipzig.asv.toolbo
    - org
    - org.json
    - te
    - te.indexer
    - te.utils
  - JRE System Library [openjdk
  - Referenced Libraries
  - .settings
  - config
  - cooccsdatabase
  - corpora
  - data
  - download
  - lib
  - logs
  - Neo4jLibs
  - Neo4jLibs322
  - output
  - resources
  - .classpath
  - .project
  - log4j.out

Problems | Javadoc | Declaration | Console | Progress

No consoles to display at this time.

Writable | Smart Insert | 1021 : 22 : 24878

3. The NLP / TM Pipeline in Hagen NLPToolbox



These steps are included in Hagen NLPToolbox! The other steps need some programming efforts (not too hard ^^).

3. Data Collection or Assembly

- Some Pointers to Useful Tools:
  - Crawler4j (Powerful open source web crawler for Java, https://github.com/yasserg/crawler4j)
  - Apache Nutch (Highly extensible and scalable open source web crawler, http://nutch.apache.org/)
  - HTTrack (Website copier, https://www.httrack.com/)
  - Selenium WebDriver (Automating web browsers, website testing, browser emulation, https://www.selenium.dev/)
  - jsoup (Java HTML Parser, https://jsoup.org/)
  - Script language Perl (Special support for regular expressions and text/string manipulation, https://www.perl.org/)

3. The NLP / TM Pipeline in Hagen NLPToolbox

- In TextProcessing.java:

1. Format conversion of files in (data/input) using Apache Tika
2. Sentence File extraction (both modes) based on language detection (LanIKernel)
3. Start of analysis method (mode 0) or creation of co-occurrence graph database (mode 1)

Touch Points for own methods / extensions

- In Cooccs.java:

4. Part-of-Speech-tagging
5. Baseform reduction
6. Stopword removal
7. Co-occurrence graph database generation using Neo4j embedded (mode 1) and creation of in-memory database (mode 0)

Helper methods for centroid calculations and query expansion based on spreading activation

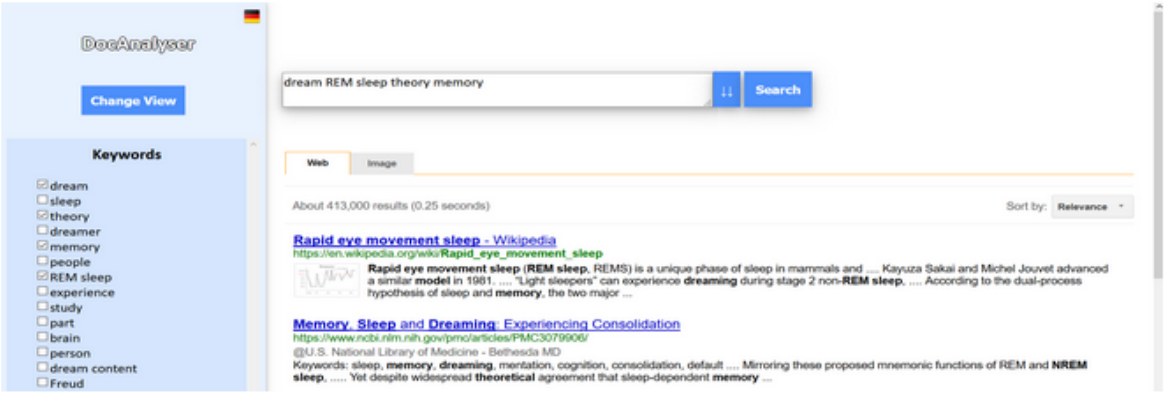
8. Determination of most important keywords using ext. PageRank, ext. HITS and centroid calculation (mode 0)
9. Output of analysis results (data/output)

3. Hagen NLPToolbox in Action (www.docanalyser.de)

DocAnalyser - Find Similar and Related Web Documents

What is DocAnalyser?

DocAnalyser is a new service that offers you novel way to **search for similar and related web documents** and to **track topics** without the need to enter search queries manually. You just need to provide a web content to be analysed. DocAnalyser then extracts its main topics and their sources (important inherent, influential aspects / basics) and uses them as search words.



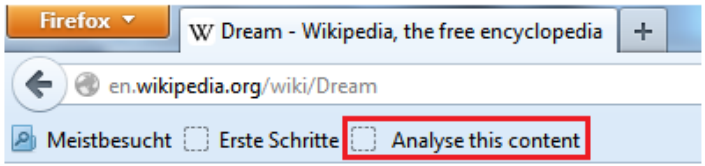
Installing DocAnalyser

In order to be able to use DocAnalyser, please **drag and drop one or both of the following bookmarklets to your bookmarks toolbar** of your favourite web browser:

Bookmarklet 1:

Analyse this content

(analyse currently shown/ selected web content)



3. Analysis of en.wikipedia.org/wiki/Systems_development_life_cycle

DocAnalyser

Change View

Keywords

- [x] system
- [x] requirement
- [x] design
- [x] development
- [ ] software
- [ ] project
- [ ] analysis
- [ ] phase
- [ ] stage
- [ ] process
- [ ] engineer
- [ ] sdlc
- [ ] test

Source Topics

- [ ] process
- [ ] sdlc
- [ ] analysis
- [ ] model
- [ ] stage
- [ ] test
- [ ] phase
- [ ] engineer
- [ ] information
- [x] development
- [ ] business
- [x] requirement
- [ ] user

system requirement design development

Search

Web Image

About 1,610,000,000 results (0.64 seconds)

Sort by: Relevance

Systems Design - Master of Science
Ad blog.fh-kaernten.at/master/**systems-design**
Development and implementation of complex **systems**. Cope with complex and cross-circular **systems**. Wide Range Of Programs. Apply Online. View Events. Highlights: A Private Non-Profit Foundation, Workshops Available.
Visit Website

Free Help Desk **System** - Free Help Desk **System**
Ad de.searchley.com/ergebnisse
Such Free Help Desk **System**, Top Ergebnisse aus dem Web. Erhalte Relevante Infos. Qualitative Ergebnisse. Leistungsstark & Einfach. Die Besten Informationen. Finde Qualitätsergebnisse. Finde Passende Resultate. Typen: Schnellsuche, Intelligentere Suche, Effiziente Suche, Mehr Finden.
Visit Website

Software Project **Design**
Ad www.i**design**.net/
Go beyond the PMP. Learn original techniques for sucessful projects. Unique in the industry. Over 15 yrs in business. Services: Software Consulting, Technology Training, Software Project **Design**.
Visit Website

Project **Development** - Project **Development**
Ad www.computerweekly.com/security
Learn how advanced AI can understand email communications & provide security.
Visit Website

Design input: What you shouldn't forget
www.johner-institute.com › Articles › Regulatory Affairs › And more ...
21 Oct 2018 ... "**Design** Input" refers to the **development** specifications, and it's not ... The term **system requirements** specification contains two separate terms:.

System Design and **Development** | The MITRE Corporation
The MITRE Corporation › publications › system-design-and-development
System design is the process of defining the components, modules, interfaces, and data for a **system** to satisfy specified **requirements**. **System development** is ...

NATO STANDARD AQAP-2110 NATO QUALITY ASSURANCE ...
www.bundeswehr.de › resource › blob › aqap-2110-2016-eng-data
File Format: PDF/Adobe Acrobat
24 Jun 2016 ... AQAP-2110. NATO QUALITY ASSURANCE. **REQUIREMENTS** FOR **DESIGN**,. **DEVELOPMENT** AND PRODUCTION. Edition D Version 1.

End of Part I

3. Practical Demonstration (Part 2)

- Storing, Analysing and Visualising Co-occurrence Graphs using Neo4j:
  - Neo4j is a (NoSQL) graph database for connected data
  - Modes: Embedded and Server
  - Neo4j Community Server 4.2.3 (installation by unpacking) and Neo4j Graph Data Science Library (GDS):
    - Go to https://neo4j.com/download-center/#community and download (depending on your system):
      1. Neo4j 4.2.3 (tar) for Linux or Neo4j 4.2.3 (zip) for Windows
      2. Neo4j Graph Data Science Library 1.5.0 (unpack the zip to find the file neo4j-graph-data-science-1.5.0.jar; also consult: https://neo4j.com/docs/graph-data-science/current/installation/)

3. Practical Demonstration (Part 2)

- Installing Neo4j Graph Data Science Library:
  - Put neo4j-graph-data-science-1.5.0.jar into the folder $NEO4J_HOME/plugins/ where $NEO4J_HOME points to the main directory of the Neo4j Community Server.
  - Configuration: Add the following lines to $NEO4J_HOME/conf/neo4j.conf :

    dbms.security.procedures.unrestricted=gds.*
    dbms.security.procedures.whitelist=gds.*

  - Test the installation:
    - Start the server via CLI by: $NEO4J_HOME/bin/neo4j console
      and in a browser: open http://localhost:7474/
    - Run the Cypher query: RETURN gds.version()
      or: CALL gds.list()

3. Practical Demonstration (Part 2)

- Installing an example co-occ. database from Hagen NLPToolbox in Neo4j Community Server:

  1. Unzip file Software_Security_Wiki_EN_cooccsdatabase.zip from the folder *corpora* .

  2. Move or copy the subdirectories *databases* and *transactions* from the folder *cooccsdatabase/data/* you just extracted to $NEO4J_HOME/data/ .

  3. Restart Neo4j Community Server.

  Note: Only one database can be active at a time when using community edition.

3. Practical Demonstration (Part 2)

- Using Neo4j Graph Data Science Library:
  - Node label: 'SINGLE_NODE' (as in Hagen NLPToolbox)
    Relationship label: 'IS_CONNECTED' (as in Hagen NLPToolbox)
  - IMPORTANT NOTE: Graph algorithms run on a graph data model which is a projection of the Neo4j property graph data model. **A graph projection can be seen as a view over the stored graph, containing only analytically relevant, potentially aggregated, topological and property information.** Graph projections are **stored entirely in-memory** using compressed data structures optimized for topology and property lookup operations.
  - Checking, if graph *my-coocc-graph* exists:
    CALL gds.graph.exists('my-coocc-graph') YIELD exists;
  - Dropping/removing the graph *my-coocc-graph* :
    CALL gds.graph.drop('my-coocc-graph') YIELD graphName;

Database Information

Use database

neo4j - default

Node Labels

*(2,512) SINGLE_NODE

Relationship Types

*(41,326) IS_CONNECTED

Property Keys

cost count dice name occur

Connected as

Username: neo4j
Roles: -
Disconnect: :server disconnect

DBMS

Version: 4.2.3
Edition: Community
Name: neo4j
Databases: :dbs
Information: :sysinfo
Query List: :queries

```
neo4j$ CALL gds.graph.exists('my-coocc-graph') YIELD exists;
```

```
neo4j$ MATCH (n) WHERE EXISTS(n.name) RETURN DISTINCT "node" ...
```

Table | Text | Code

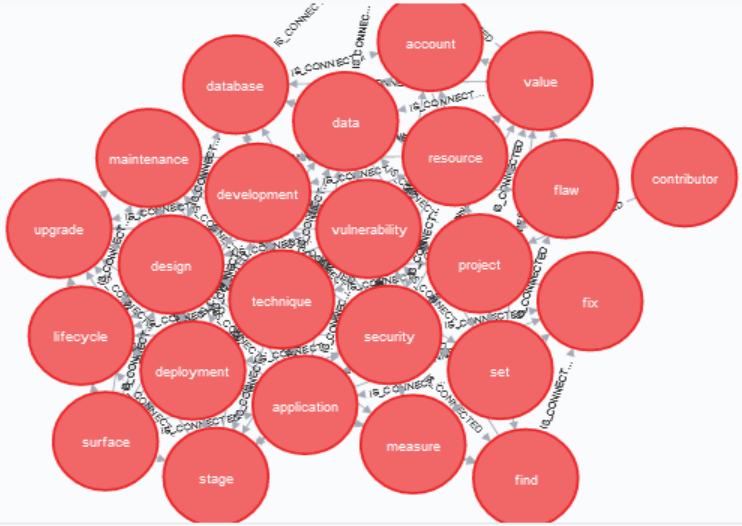
	entity	name
1	"node"	"application"
2	"node"	"security"
3	"node"	"contributor"

Started streaming 3 records after 2 ms and completed after 66 ms.

```
neo4j$ MATCH (n) RETURN n LIMIT 25
```

Graph | Table | Text | Code

*(25) SINGLE_NODE(25)
*(150) IS_CONNECTED(150)



Displaying 25 nodes, 150 relationships.

3. Practical Demonstration (Part 2)

- Using Neo4j Graph Data Science Library:
  - Creating a graph from the example co-occurrence database:

```
CALL gds.graph.create(
    'my-coocc-graph',
    'SINGLE_NODE',
    'IS_CONNECTED',
    {
        relationshipProperties: ['dice','cost']
    }
)
YIELD graphName, nodeCount, relationshipCount, createMillis;
```

3. Practical Demonstration (Part 2)

- Using Neo4j Graph Algorithms:
  - PageRank of Nodes:

```
CALL gds.pageRank.stream('my-coocc-graph‘, {
        relationshipWeightProperty: 'cost'})
YIELD nodeId, score
RETURN gds.util.asNode(nodeId).name AS name, score
ORDER BY score DESC, name ASC
LIMIT 250
```

  - Clustering Nodes:

```
CALL gds.labelPropagation.stream('my-coocc-graph’, {
        relationshipWeightProperty: 'cost’})
YIELD nodeId, communityId AS Community
RETURN gds.util.asNode(nodeId).name AS Name, Community
ORDER BY Community, Name
```

3. Practical Demonstration (Part 2)

- Using Neo4j Graph Data Science Library:
  - Shortest Distance of Nodes:

```
MATCH (source:SINGLE_NODE {name: 'software'}), (target:SINGLE_NODE
{name: 'attack‘})

CALL gds.beta.shortestPath.dijkstra.stream('my-coocc-graph‘, {
sourceNode: id(source),
targetNode: id(target),
relationshipWeightProperty: 'cost‘})

YIELD index, sourceNode, targetNode, totalCost, nodeIds, costs
RETURN
index,
gds.util.asNode(sourceNode).name AS sourceNodeName,
gds.util.asNode(targetNode).name AS targetNodeName,
totalCost,
[nodeId IN nodeIds | gds.util.asNode(nodeId).name] AS nodeNames,
costs
ORDER BY index
```

4. Recommended Alternative Libraries (mostly Java-based)

- GATE (https://gate.ac.uk/ , most comprehensive)
- Stanford CoreNLP (https://stanfordnlp.github.io/CoreNLP/)
- LingPipe (http://www.alias-i.com/lingpipe/index.html)
- Deeplearning4j (https://deeplearning4j.org/)
- Apache Spark with MLlib (https://spark.apache.org/mllib/)
- Apache OpenNLP (https://opennlp.apache.org/ ,)

- Python-based libraries: spaCy (operates with TensorFlow), Gensim (topic modelling, word embeddings), NLTK (toolkit with longest history)

4. Featured Alternative Apache OpenNLP

- Robust NLP Library Apache OpenNLP (https://opennlp.apache.org/):
  - Actively cared for
  - Supports all mentioned preprocessing steps
  - Comes along with language specific models and resources for these tasks (http://opennlp.sourceforge.net/models-1.5/)
  - Also supports tasks such as syntactic parsing, named entity extraction and coreference resolution
  - Full documentation (Javadoc, manual and Wiki) at: https://opennlp.apache.org/docs/

One more thing ^^
Demo-App OpenNLPTest

(Eclipse-Project, 22 MB)

https://www.mario-kubek.de/projects/OpenNLPTest.7z

https://github.com/drmakube/OpenNLP_TestApp

5. Summary

- Hagen NLPToolbox (March 2021 edition) presented
- Discussed the pros and cons
- Practical demonstration in two parts
- Many other tools and resouces exist (Python libraries currently most successful)
- Apache OpenNLP featured and Demo-App provided

Lectures and Other Links

- My 3-day course at KMUTNB on Graph-based NLP, TM and Search Support from 2019 (also on ext. PageRank, ext. HITS, assoc. analysis, Centroid concept, WebEngine):
  - https://www.mario-kubek.de/lectures/KMUTNB_AS_Lecture_Feb2019.zip
  - https://www.mario-kubek.de/lectures/KMUTNB_AS_Lecture_Materials_Feb2019.zip
- My lecture on data preparation in automatic text processing from NLIR 2018:
  - https://www.mario-kubek.de/lectures/NLIR_Data Preparation in Automatic Text Processing.pdf
- Book: Rada Mihalcea and Dragomir Radev, Graph-based Natural Language Processing and Information Retrieval, 1st edition, Cambridge University Press, April 2011

Literature on Neo4j

- More Information on Neo4j Graph Data Science Library (and Neo4j & Cypher in general):
  - Consult the GDS manuals (also on NLP-related content):
    https://neo4j.com/docs/graph-data-science/current/
    https://neo4j.com/developer/graph-data-science/
    https://neo4j.com/developer/graph-data-science/nlp/
    https://neo4j.com/developer/graph-data-science/graph-embeddings/
  - The Neo4j Cypher Manual:
    https://neo4j.com/docs/cypher-manual/current/
  - New and free Books:

    https://neo4j.com/books/
    https://neo4j.com/graph-databases-for-dummies/
    https://neo4j.com/graph-data-science-for-dummies/
    https://neo4j.com/graph-algorithms-book/