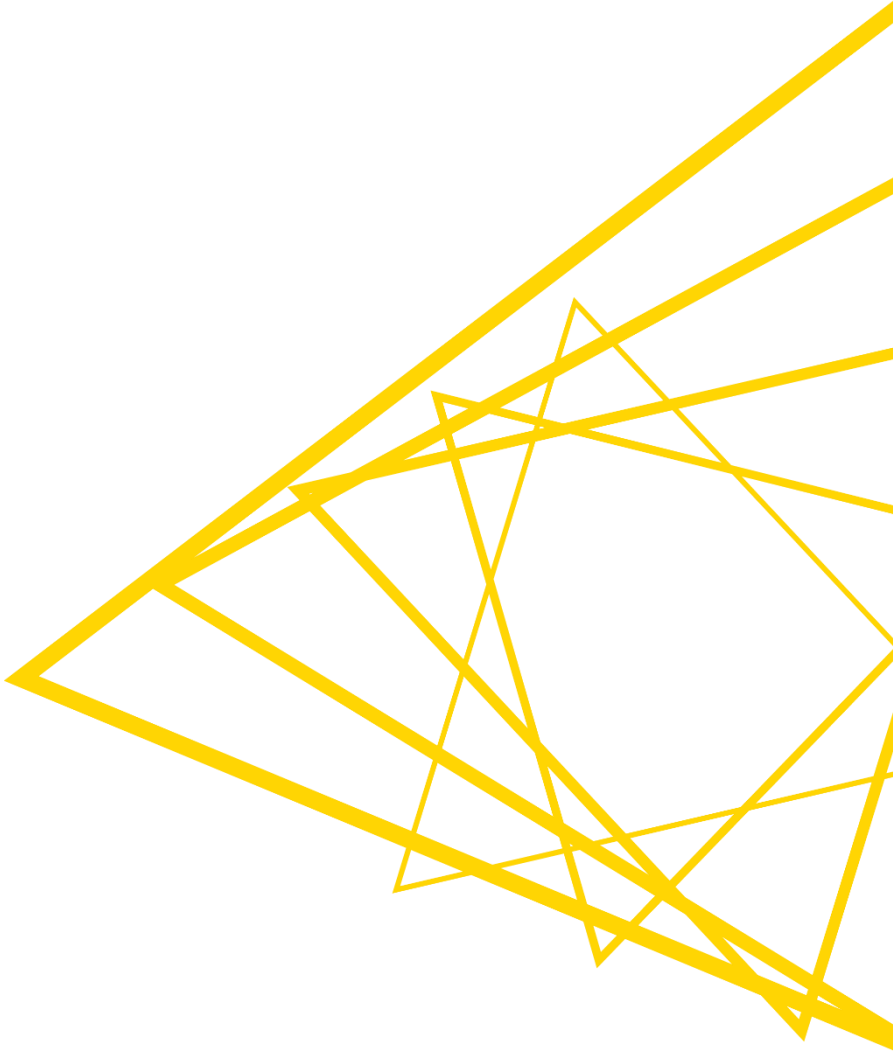
Open for Innovation
KNIME

NER Modeling and Co-occurrence networks

Julian Bunzel
julian.bunzel@knime.com

Agenda

- Use Case
- Document Collection
- NER Modeling
- Extracting Named-Entities
- Network Creation / Interactive views
- Subgraph Extraction



Use Case

- Process:
  - Train a NER model to recognize **drug names** in literature
  - Create an entity co-occurrence network
  - Predict purpose of drugs based on node neighborhood
  - Extract, visualize and validate interesting subgraphs

- Requirements:
  - Document collection to train a NER model and to extract named-entities from

Collection of Documents

- To train a model annotated documents are required
- Manual annotation is very time-consuming
- Automatic annotation is fast, but not always correct due to ambiguity etc.

In this workshop we will collect documents from **PubMed** and annotate them automatically.

Create a Dictionary of Entities

- As dictionary we used drugs covered by the WHO's Anatomical-Therapeutic-Chemical (ATC) Classification System
- ~800 drugs and drug combinations

Row ID	S Drug Name	S ATC Code
Row0	4-aminosalicylic acid	J04AA01
Row1	abacavir	J05AF06, J05AR02, J05AR13
Row2	abatacept	L04AA24
Row3	abciximab	B01AC16
Row4	abemaciclib	L01XE50
Row5	abiraterone	L02BX03
Row6	aciclovir	A16AX08
Row7	adalimumab	L04AB04
Row8	adefovir dipivoxil	J05AF08
Row9	afamelanotide	D02BB02
Row10	afatinib	L01XE13
Row11	aflibercept	L01XX44, R03BB05, S01LA05
Row12	agalsidase alfa	A16AB03
Row13	agalsidase beta	A16AB04
Row14	agomelatine	N06AX22

ATC Code

acetylsalicylic acid

N02BA01

ATC Code - Level 1

acetylsalicylic acid

N02BA01

Anatomical maingroup: N = Drugs for nervous system

ATC Code - Level 2

acetylsalicylic acid

N02BA01

Therapeutic maingroup: N02 = Analgenics/Painkiller

ATC Code - Level 3

acetylsalicylic acid

N02BA01

Therapeutic/pharmacological subgroup: N02B = Other analgesics and antipyretics

ATC Code - Level 4

acetylsalicylic acid

N02BA01

Therap./pharmakol./chemical subgroup: N02BA = Salicylic acid and derivatives

ATC Code - Level 5

acetylsalicylic acid

N02BA01

Substance name: N02BA01 = Acetylsalicylic acid

Collecting Documents

- Collecting a set of documents containing dictionary entities from PubMed.
- PubMed is an openly-available search engine for biomedical literature with more than 29 million entries
- For each dictionary entry PubMed was queried using the Document Grabber node (max. 100 results per query)
- Collected altogether approx. 72.000 documents

Preparing Documents for Training

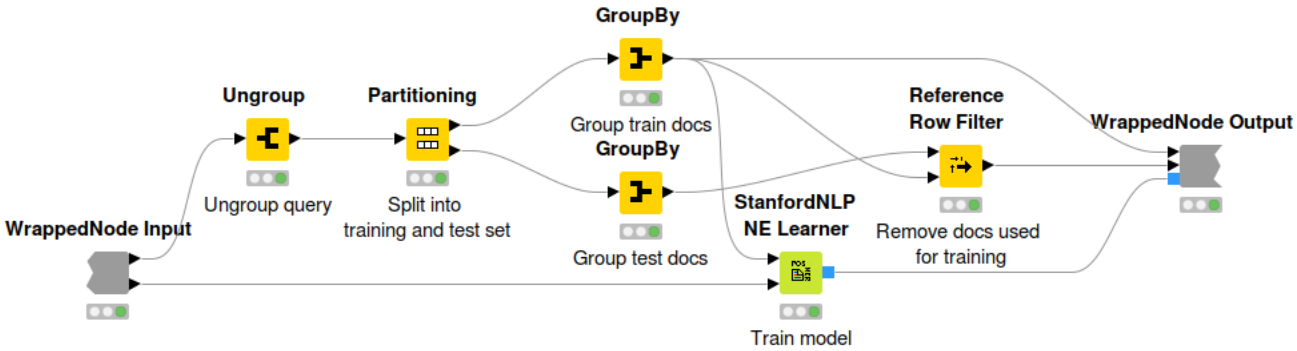
- Removing drug names with <20 hits on PubMed
- Removing documents that do not contain the exact query term (only related words)

→ Ensuring enough sample sentences for each drug name

→ Reduced set of documents to approx. 45.000 *unique* documents

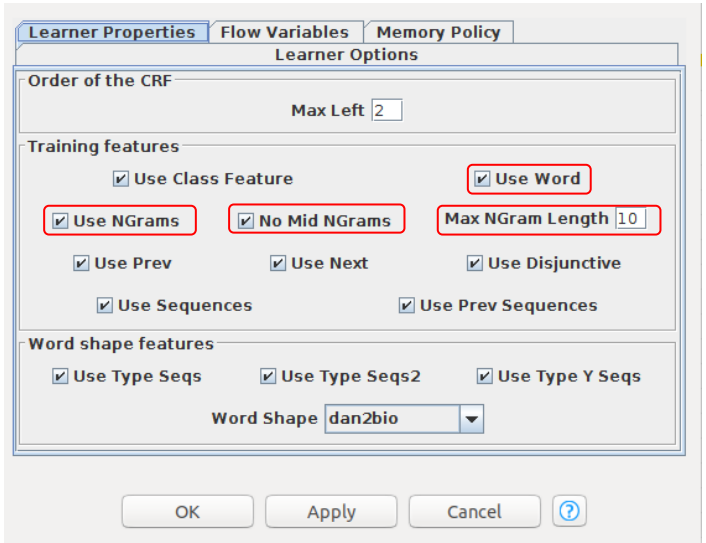
Train NER Model – StanfordNLP NE Learner

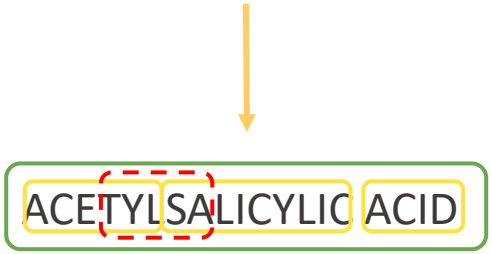
- Training by using the StanfordNLP NE Learner node
  - Top port: The (training) documents
  - Bottom port: Dictionary of drug names
- Internally annotates the documents based on the dictionary
- Linear-chain conditional random field (CRF) model
- Split of document collection into training and test set (10% / 90%)



StanfordNLP NE Learner - Dialog

- Use Word: Whole entity is used as feature
- Use NGrams: Substrings of a word are used as feature
- No Mid NGrams: Only use substrings that include the beginning or ending of a word
- Max NGram Length: Maximum length of a substring





Evaluating the model – StanfordNLP NE Scorer

- Evaluating the model with StanfordNLP NE Scorer node
  - Top Port: the (test) documents
  - Bottom Port: the model
- Annotation of documents in two different ways
  - Regular expressions based on dictionary
  - Trained NER model
- Calculating statistics such as precision and recall

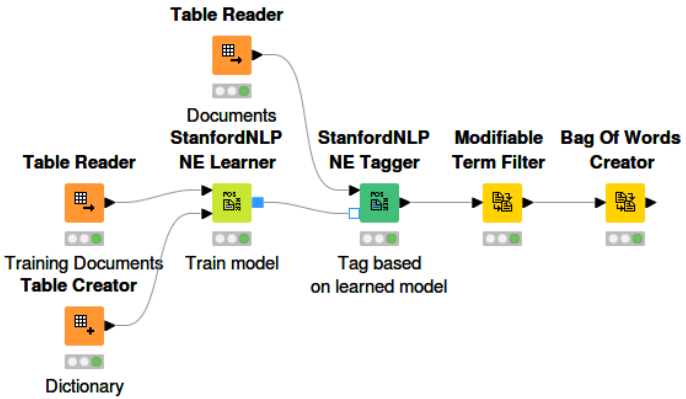
Row ID	D Precision	D Recall	D F1	I TP	I FP	I FN
Row0	0.983	0.99	0.987	179668	3110	1773

S Entities	S Regex anno.	S Model anno.
INSULIN	DRUG	DRUG
INSULIN ASPART	DRUG	O
INSULIN ISOPHANE	O	DRUG

- Possible drawbacks
  - Model is trained to detect entirely new entities
  - Bias in *False Positives* and *False Negatives*

Annotation and Extraction of Named-Entities

- Annotating documents with StanfordNLP NE Tagger
- Annotations in total: 184.551
- Unique entities: 1.531
  - Entities from dictionary: 731
  - Entirely new entities: 800



Term
fenfluramine[DRUG(PHARMA)]
dexfenfluramine[DRUG(PHARMA)]
benfluorex[DRUG(PHARMA)]
Sipuleucel-T[DRUG(PHARMA)]
Ipilimumab[DRUG(PHARMA)]
delamanid[DRUG(PHARMA)]
betahstine[DRUG(PHARMA)]
betahistine[DRUG(PHARMA)]
Betahistin[DRUG(PHARMA)]
nalmefene[DRUG(PHARMA)]
imiquimod[DRUG(PHARMA)]
riociguat[DRUG(PHARMA)]

Open for Innovation KNIME

Annotation and Extraction of Named-Entities

- Annotating documents with StanfordNLP NE Tagger
- Annotations i
- Unique entitie
  - Entities fr
  - Entirely n

Examples for new entities:

INSULIN

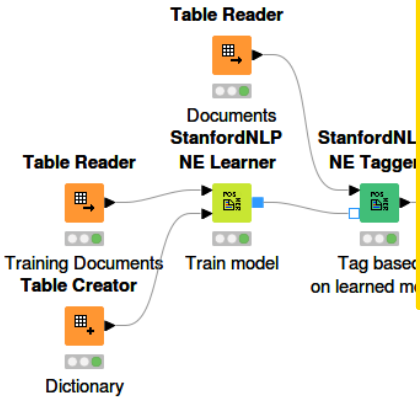
INSULIN DEGLUDEC

INSULIN ISOPHANE

BASEDOXIFENE

LASOFOXIFENE

PIPENDOXIFENE



rm
ramine[DRUG(PHARMA)]
nfluramine[DRUG(PHARMA)]
orex[DRUG(PHARMA)]
ucel-T[DRUG(PHARMA)]
umab[DRUG(PHARMA)]
anid[DRUG(PHARMA)]
stine[DRUG(PHARMA)]
istine[DRUG(PHARMA)]
istin[DRUG(PHARMA)]
efene[DRUG(PHARMA)]
mod[DRUG(PHARMA)]
uat[DRUG(PHARMA)]

Open for Innovation KNIME

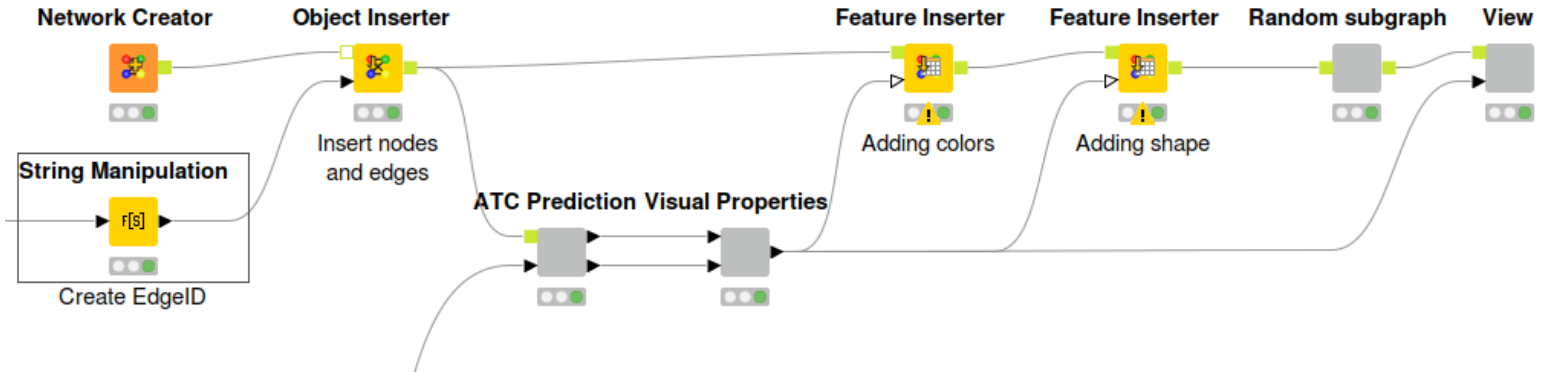
Creating a co-occurrence network

- Counting co-occurrences of named entities using Term Co-occurrence Counter
- For each named-entity a node will be created
- Co-occurring named-entities will be connected by an edge
- Features to be visualized in network:
  - First level of ATC code as *node color*
  - Entities from dictionary as *circles*
  - Additionally recognized entities as *squares*

Term1	Term2	Document ...
bortezomib[DRUG(PHARMA)]	trastuzumab[DRUG(PHARMA)]	1
carfilzomib[DRUG(PHARMA)]	trastuzumab[DRUG(PHARMA)]	1
bortezomib[DRUG(PHARMA)]	carfilzomib[DRUG(PHARMA)]	1
carfilzomib[DRUG(PHARMA)]	ixazomib[DRUG(PHARMA)]	1
bortezomib[DRUG(PHARMA)]	ixazomib[DRUG(PHARMA)]	1

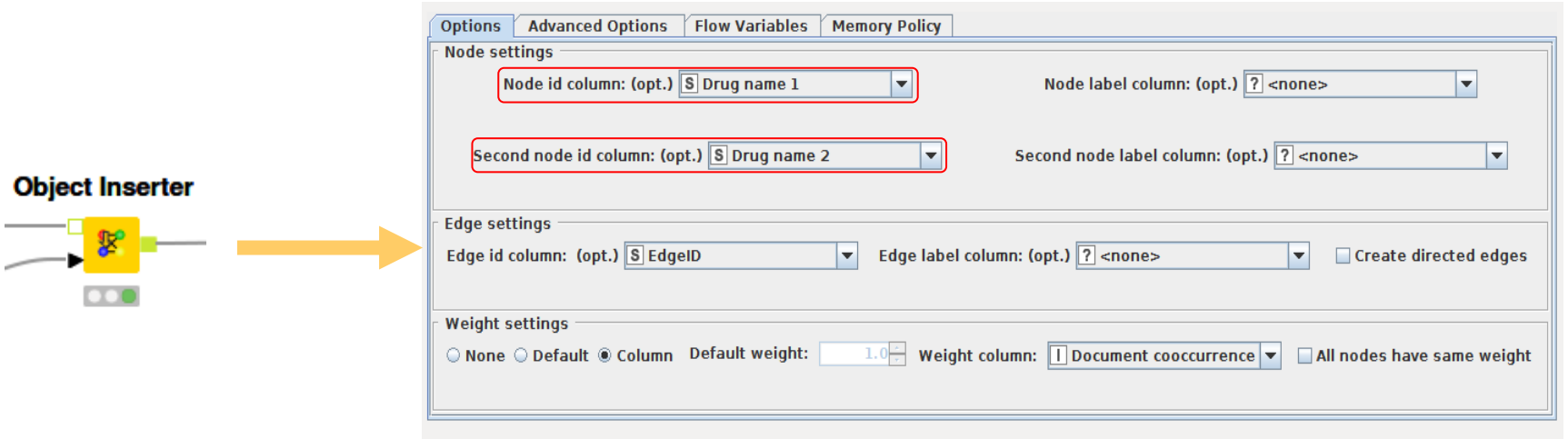
Creating a co-occurrence network

- Nodes to create the network
  - Network Creator node
  - Object Inserter node
  - (Multi) Feature Inserter node
  - Network Viewer (JavaScript)



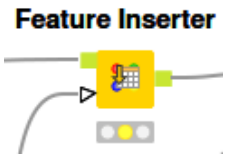
Object Inserter

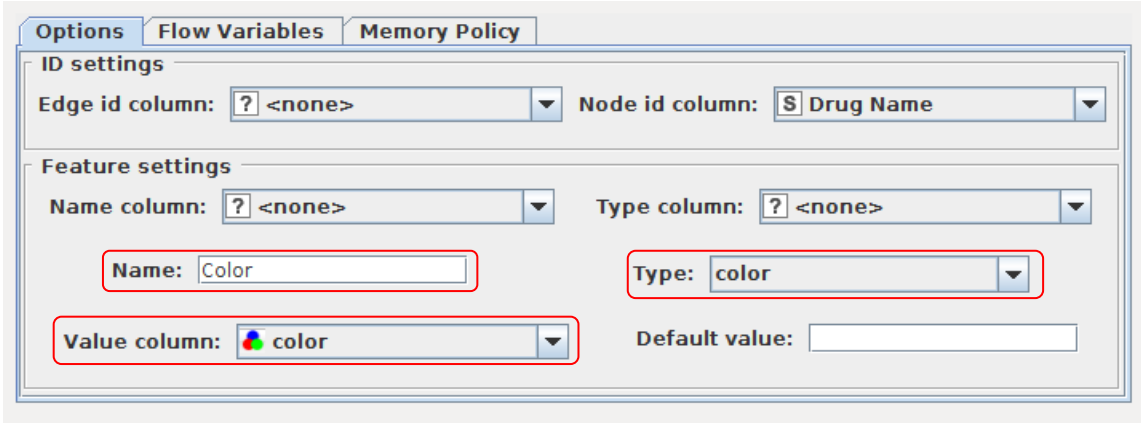
- Used to add nodes and edges to a network
- Node id column: Column of named-entities
- Second node id column: Co-occurring named-entities
- Edges are set automatically



Feature Inserter

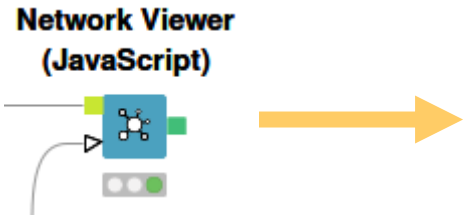
- Used to set (visual) properties for a network
- Name: name of property (can be selected by Viewer nodes)
- Type: type of property
- Value column: Column of previously defined colors to set for nodes or edges

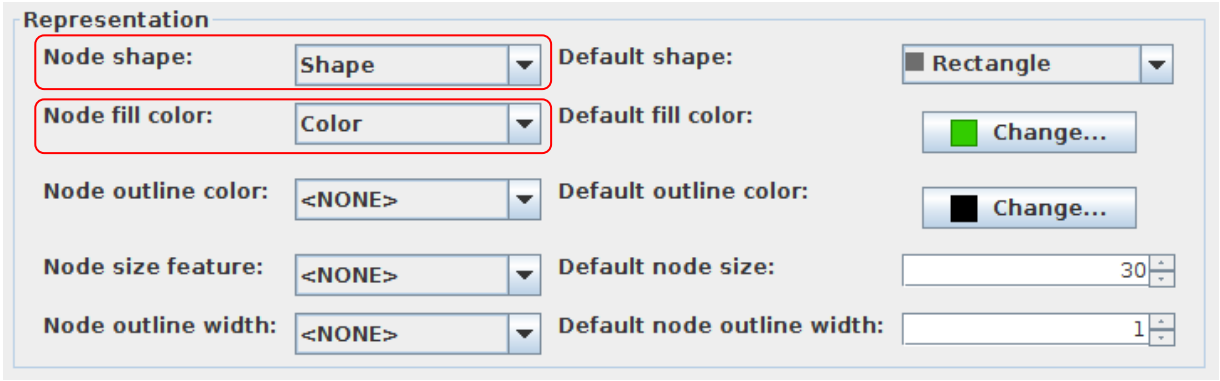




Network Viewer (JavaScript)

- Provides different algorithms to layout the network
  - Can be selected interactively in the view
- Set node shape, color, outline color etc.
- Possibility to generate an image

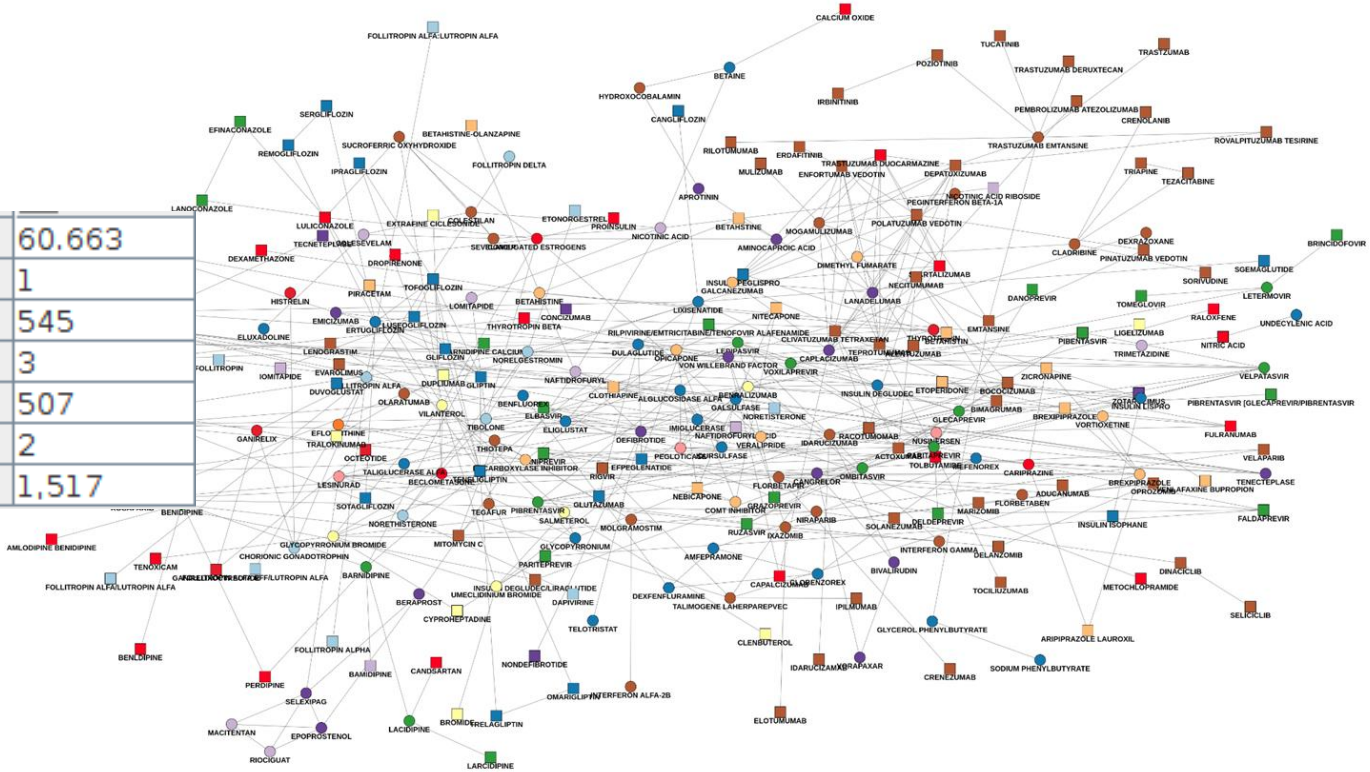




Named-Entity Co-Occurrence Network

- Final network consist of 1.521 nodes and 46.134 edges

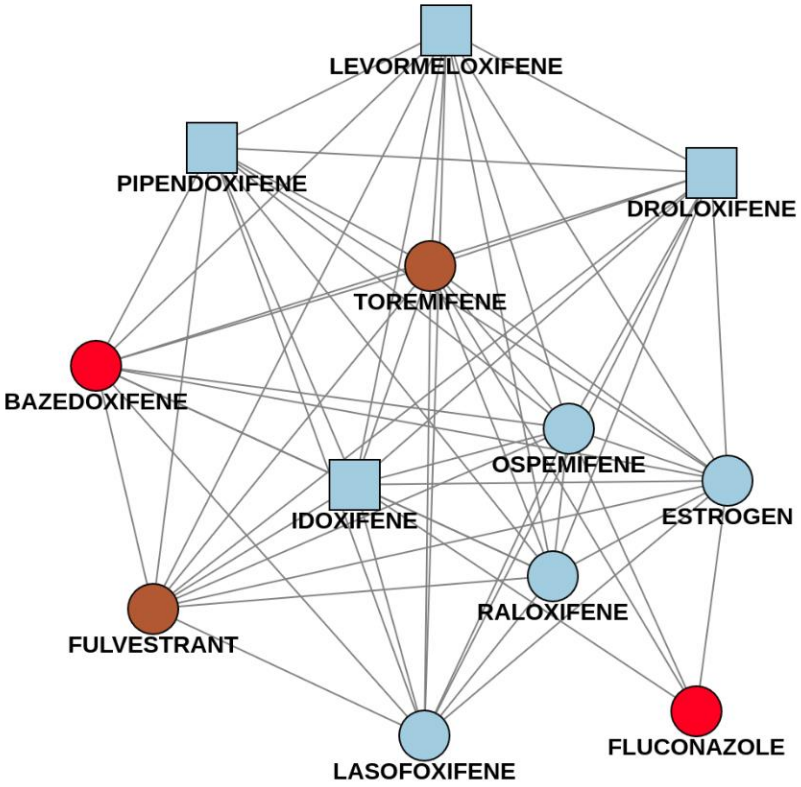
Avg. node degree	60.663
Min. node degree	1
Max. node degree	545
No. of components	3
Avg. component size	507
Min. component size	2
Max. component size	1,517

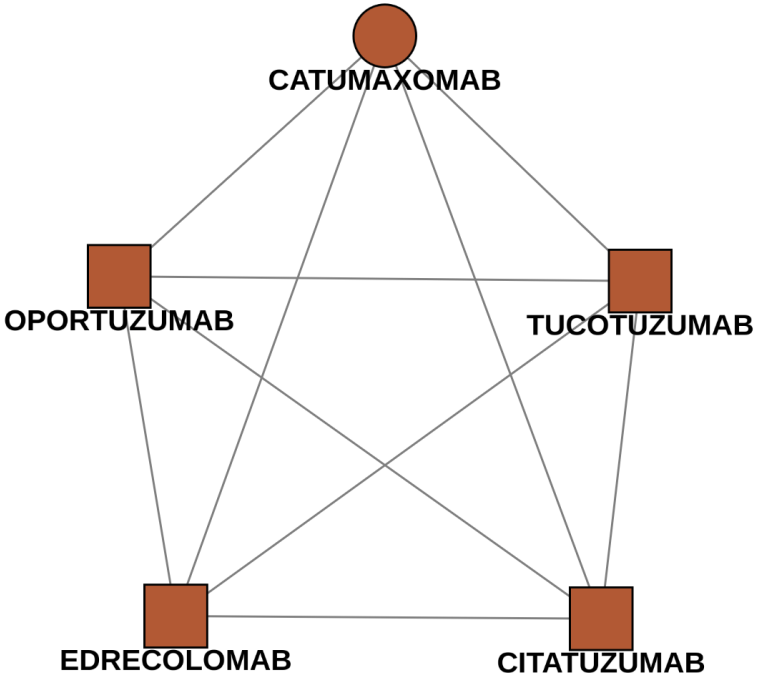


Extracting subgraphs

- Aim is to extract subgraphs to validate ATC prediction
- Extracting connected components of newly identified named-entities and their first neighborhood of known named-entities

Subgraphs





Summary

- What we learned:
  - Training an NER model
  - Creating a co-occurrence network
  - Extracting and visualizing subgraphs

- **Note: This can be applied to any other domain**

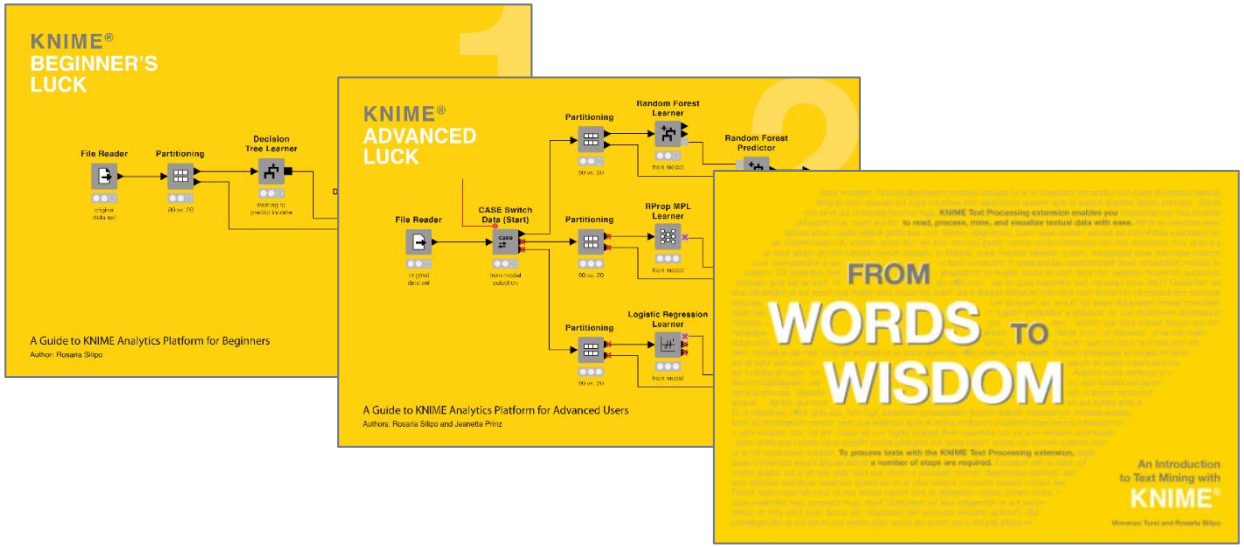
KNIME Books

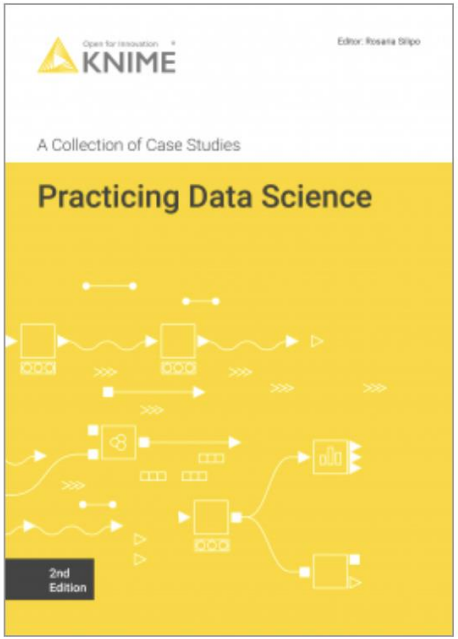
Course books are available on **KNIME Press**

https://www.knime.com/knimepress

Download your free copy by using the code:

FALL-SUMMIT-WORKSHOP





Open for Innovation KNIME

Open for Innovation
KNIME

Thank you for joining!