Open for Innovation
KNIME

KNIME User Group Meeting 2015
Text Mining Workshop

Kilian Thiel
Kilian.Thiel@knime.com
@KNIME

Agenda

- Introduction
- Sentiment Analysis
  - Single word features
  - Ngram features
  - Dictionary based

Resources

- **The KNIME Website** (www.knime.org)
  - **Textprocessing** (tech.knime.org/knime-text-processing)
  - **Forum** (tech.knime.org/forum/knime-textprocessing)

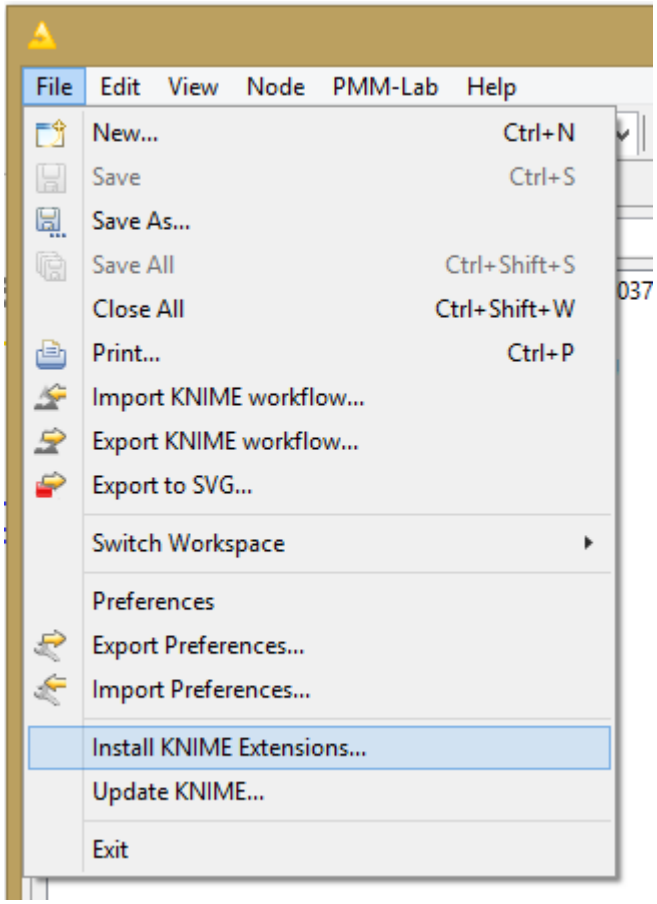
  - **Blog** for news, tips and tricks(www.knime.org/blog)

- **KNIME TV** channel on YouTube
- **KNIME** on **@KNIME**

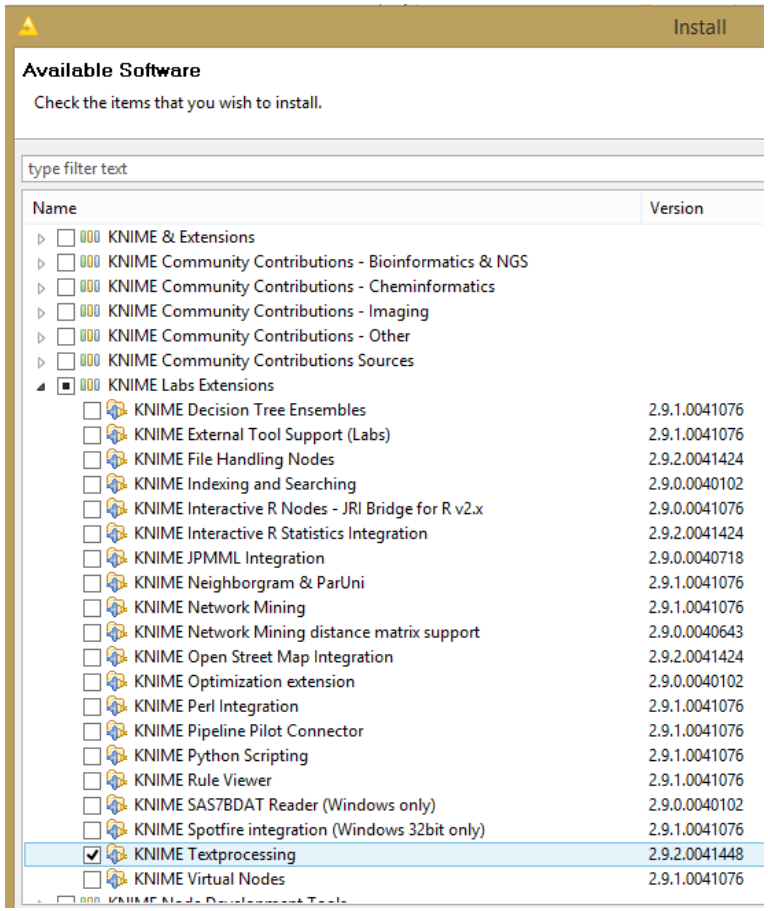
Introduction to KNIME Textprocessing

Installation

1.)



2.)

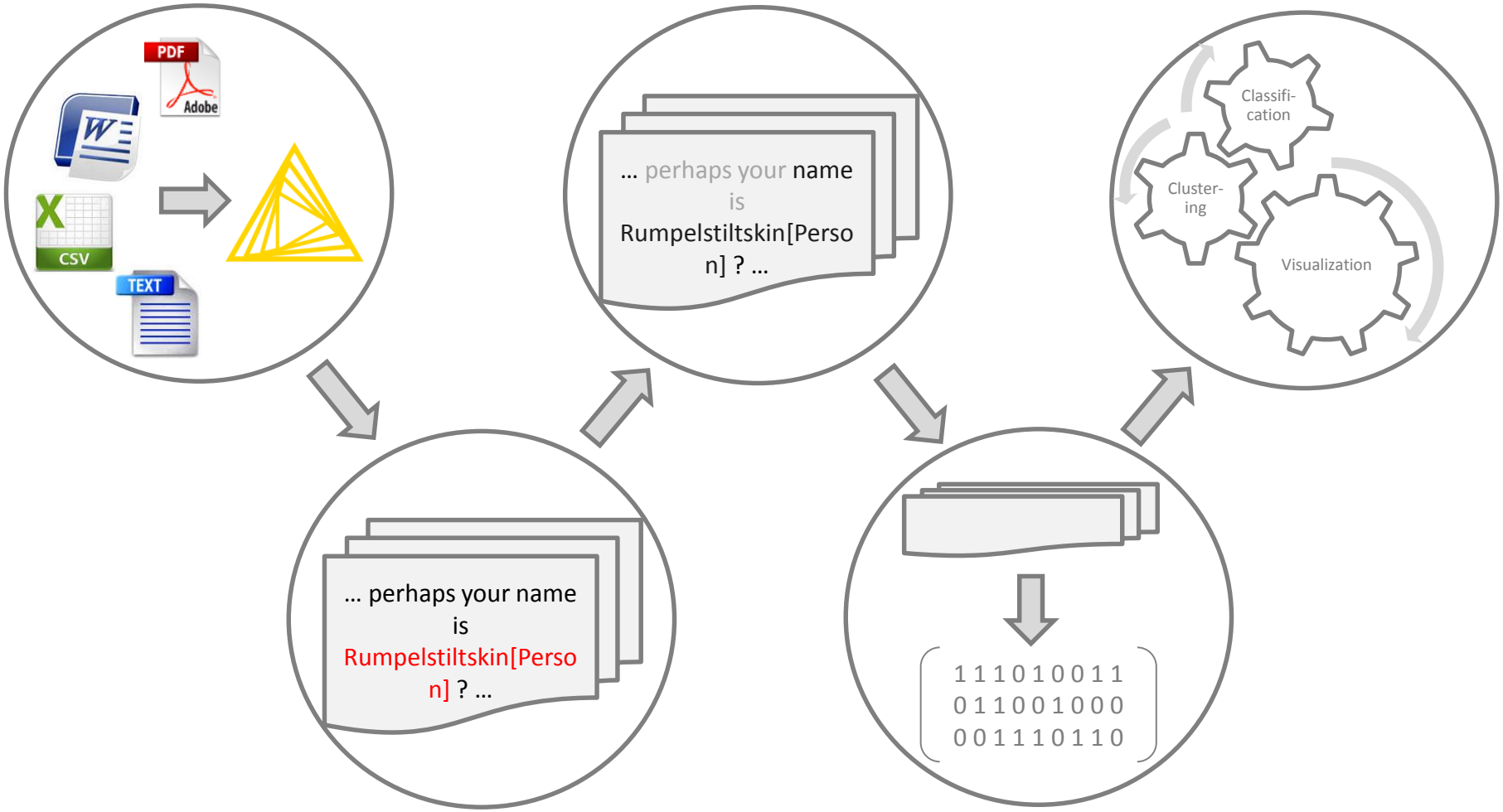


Tip

- Increase maximum memory for KNIME
- Edit knime.ini
  - Add "-Xmx3G" as last line of knime.ini file
  - Or change existing setting

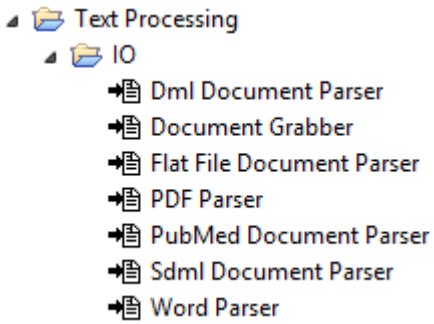
- Useful additional extensions
  - Palladian (community extension)
    - Web crawling, Text Mining
  - XML-Processing (KNIME extension)
    - Parsing and processing of XML documents

Philosophy



Data Import

- Node Repository: KNIME Labs/Text Processing/IO
- Available Parser Nodes
  - Flat File Document Parser
  - PDF Parser
  - Word Parser
  - Document Grabber
  - …

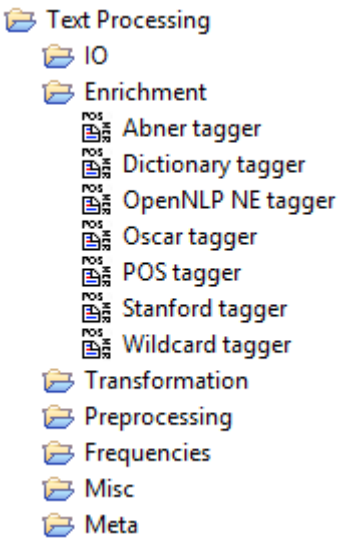


Enrichment

- Node Repository:

  KNIME Labs/Text Processing/Enrichment

- Available Tagger Nodes
  - Standford tagger
  - Dictionary (& Wildcard) tagger
  - OpenNLP tagger
  - Abner tagger
  - …

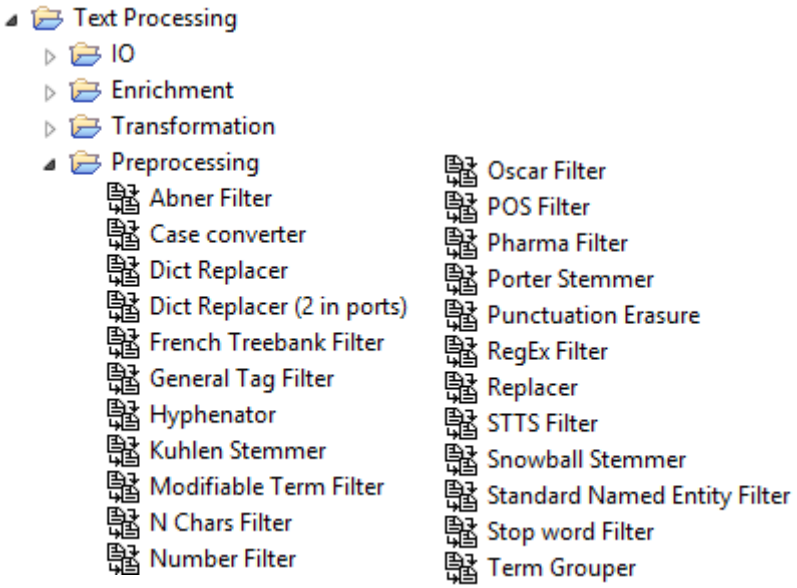


Preprocessing

- Node Repository:

  KNIME Labs/Text Processing/Preprocessing

- Available Preprocessing Nodes
  - Stop word Filter
  - Snowball Stemmer
  - General Tag Filter
  - Case converter
  - RegEx Filter
  - …

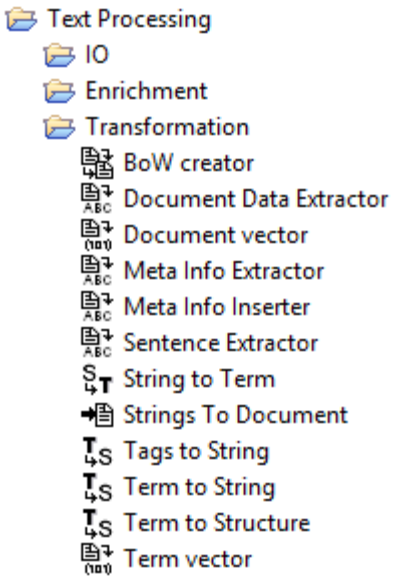


Transformation

- Node Repository:

  KNIME Labs/Text Processing/Transformation

- Available Transformation Nodes
  - BoW Creator
  - Document Vector
  - Strings to Document
  - Sentence Extractor
  - Document Data Extractor
  - …



Additional Data Types

- Document Cell
  - Encapsulates a document
    - Title, sentences, terms, words
    - Authors, category, source
    - Generic meta data (key, value pairs)
- Term Cell
  - Encapsulates a term
    - Words, tags

Document
"Great food , interesting service"
"Excellent Lunch Destination"
"Hidden treasure near KaDaWe"
"Excellent Food Very Reasonable !"
"Good food , great prices !"
"Nice food at a reasonable price"

Term
Very[RB(POS)]
good[JJ(POS)]
Thai[NNP(POS)]
food[NN(POS)]
![SYM(POS)]
Been[NNP(POS)]
there[EX(POS)]
on[IN(POS)]
Monday[NNP(POS)]
and[CC(POS)]
had[VBD(POS)]
a[DT(POS)]
great[JJ(POS)]
time[NN(POS)]

Data Table Structures

- Document table
  - List of documents

Table "default" - Rows: 440 | Spec - Column: 1 | Properties | Flow Variables

Row ID	Document
Row0_1_Row...	"Great food , interesting service"
Row0_2_Row...	"Excellent Lunch Destination"
Row0_3_Row...	"Hidden treasure near KaDaWe"
Row0_4_Row...	"Excellent Food Very Reasonable !"
Row0_5_Row...	"Good food , great prices !"

- Bag of words
  - Tuples of documents and terms

Table "default" - Rows: 21290 | Spec - Columns: 2 | Properties | Flow Variables

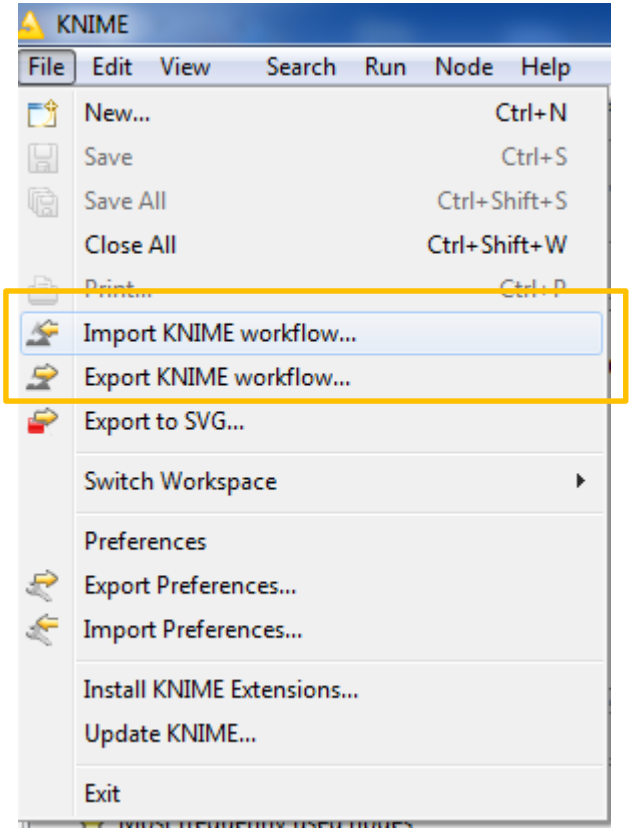
Row ID	Term	Document
Row7804	Nice[NNP(POS)]	"Nice Thai Food"
Row7805	Thai[NNP(POS)]	"Nice Thai Food"
Row7806	Food[NNP(POS)]	"Nice Thai Food"
Row7807	Good[NNP(POS)]	"Nice Thai Food"
Row7808	food[NN(POS)]	"Nice Thai Food"

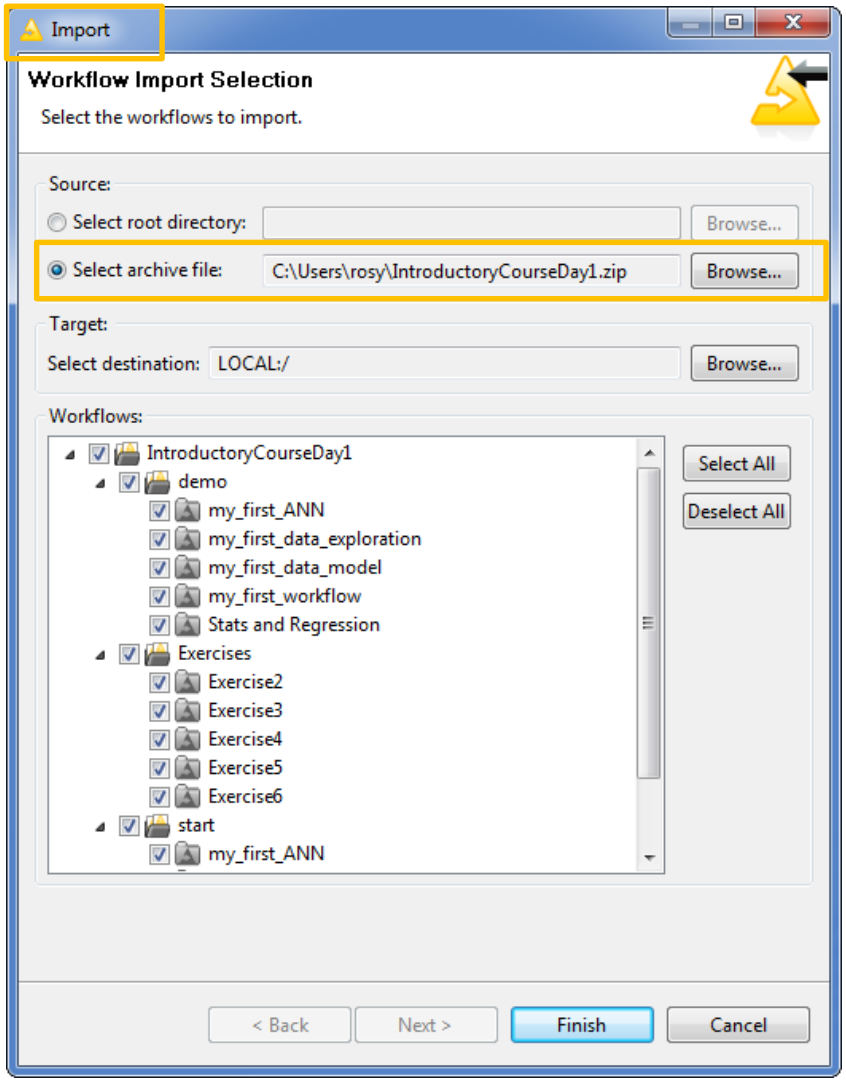
- Document vectors
  - Numerical representations of documents

Table "default" - Rows: 440 | Spec - Columns: 2613 | Properties | Flow Variables

Row ID	Document	afford	truly	elegant	restaur...
1	"...if you can afford...	1	1	1	1
2	"2 michelin stars wel...	0	0	0	1
3	"A Michelin Star exp...	0	0	1	1
4	"A Unique Place to ...	0	0	0	0
5	"A fun place to hav...	0	0	0	1

Import Export Workflows





Today‘s example

- Classification of free-text documents is a common task in the field of text mining.
- It is used to categorize documents, i.e. assigning pre-defined topics or for sentiment analysis.
- Today we construct a workflow that reads and preprocesses text documents, transforms them into a numerical representation and build a predictive model to assign pre-defined sentiment labels to documents.

Today‘s example

- Subset(s) of the Large Movie Review Dataset v1.0
- The LMR Dataset v1.0 contains 50.000 English movie reviews with their associated sentiment label “positive” or “negative”
- http://ai.stanford.edu/~amaas/data/sentiment/

Today‘s example

Goal:

- Build **classifier** to distinguish between positive and negative movie reviews.

- *„Absolute masterpiece of a film! Goodnight Mr.Tom has swiftly become one of my favourite films of all time. Nobody should miss out on seeing this film, it's just too good! …“*

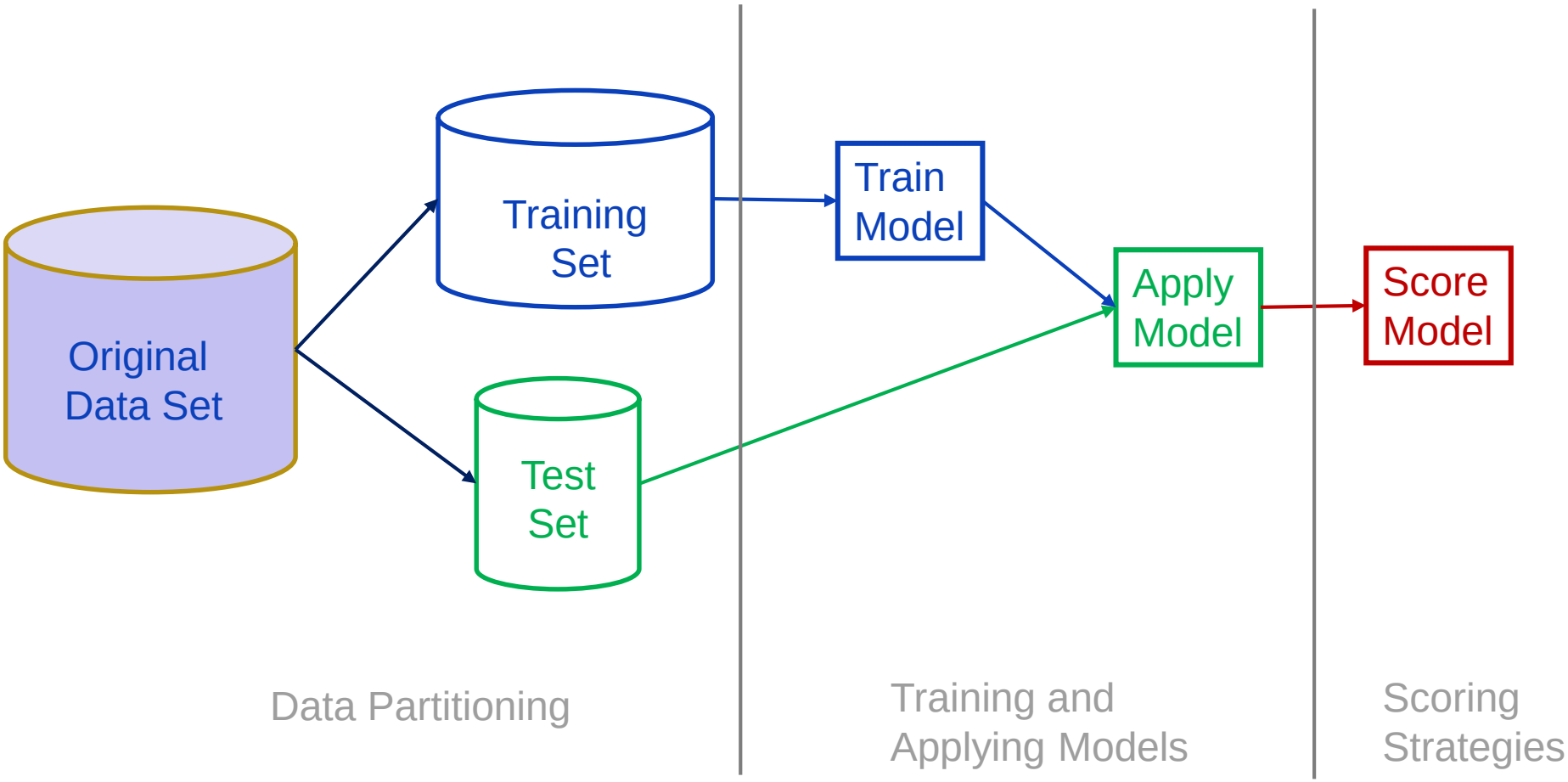
Positive or negative movie review?

Hands On

- Adjust Xmx setting in knime.ini
- Start KNIME
- Import Workflows

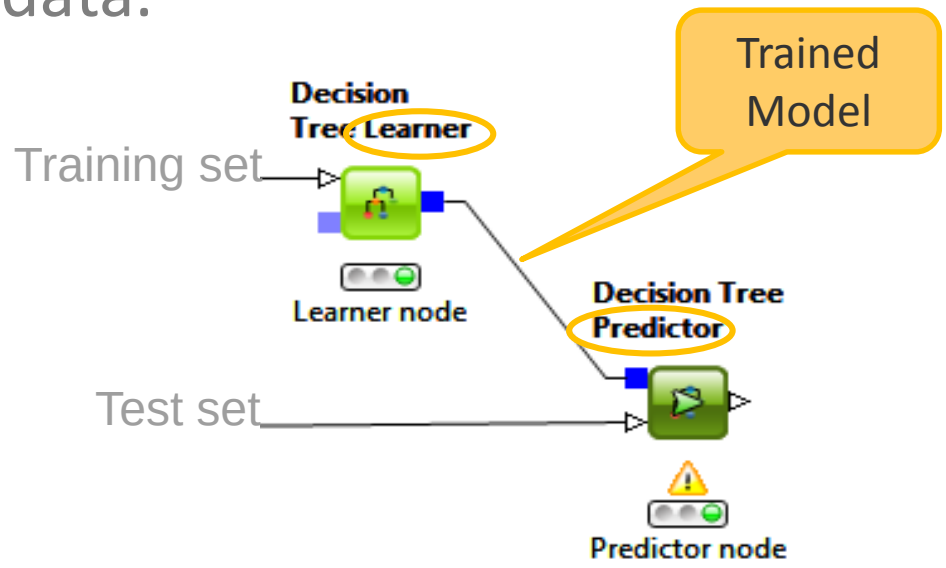
Sentiment Analysis

Classification (Recap)

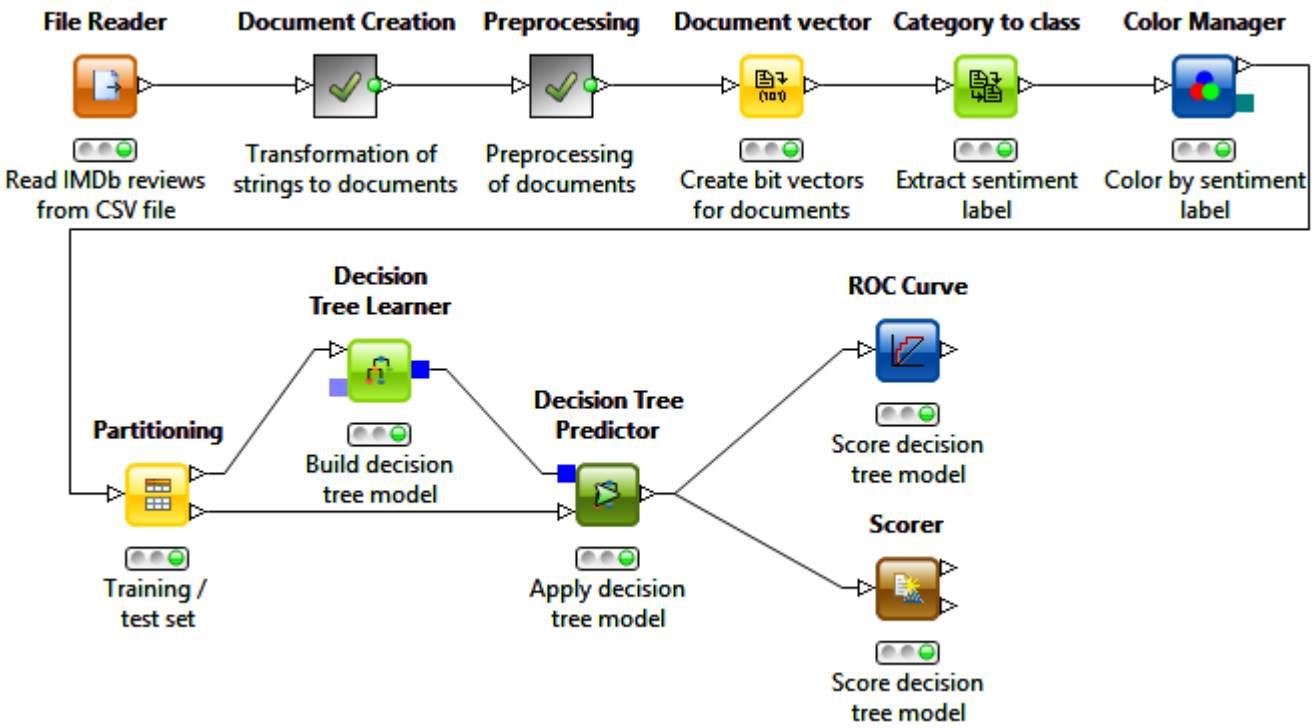


Classification (Recap)

- All data mining models use a Learner-Predictor motif.
- The Learner node trains the model with its input data.
- The Predictor node applies the model to a different subset of data.



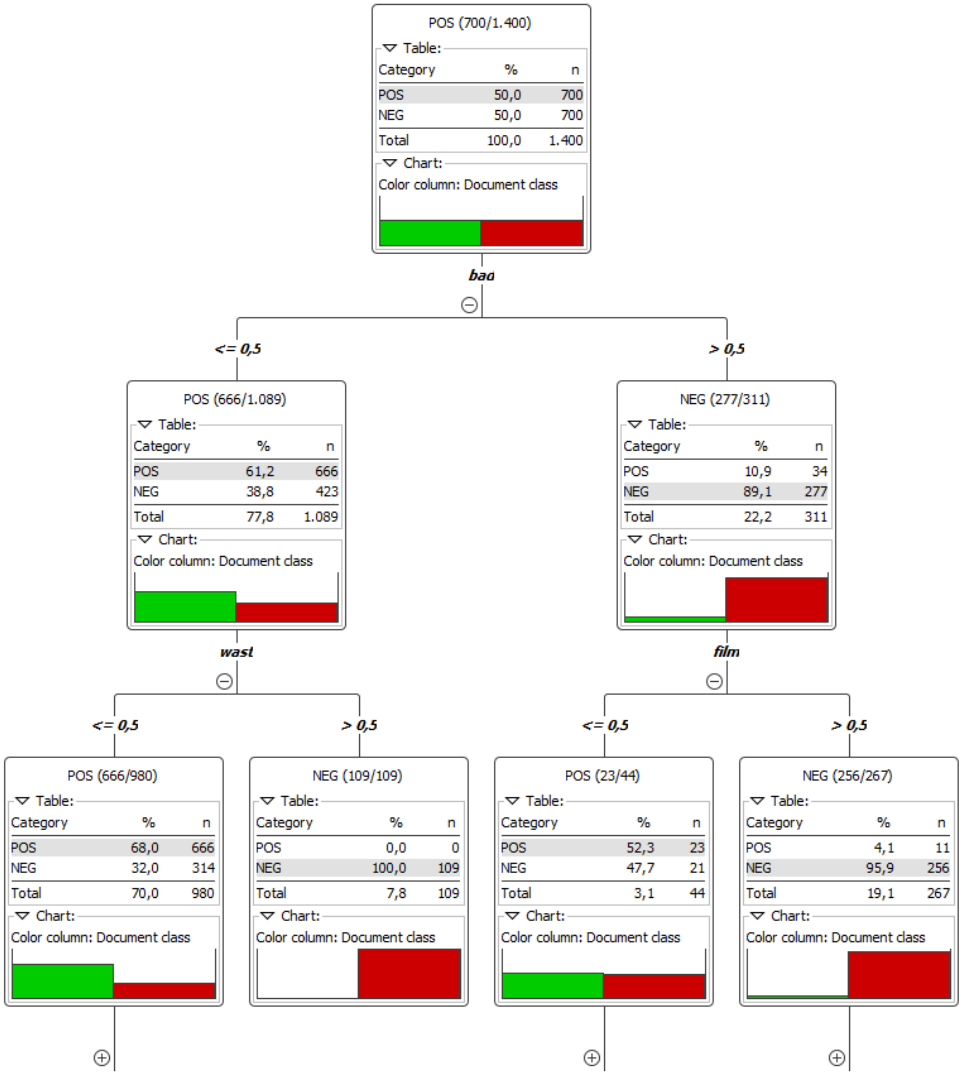
From Strings to Numbers



Hands On

- Read movie review data
- Convert to documents
- Preprocess documents
- Filter features based on frequency
- Transform documents into numerical representation
- Train and score predictive model
- View decision tree view

Decision Tree Model

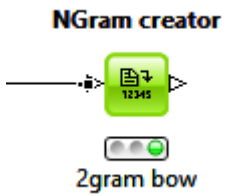


Negations

- What about negations?
  - Not good
  - Not bad
- Can not be handled with single word features
- Ngram features (1- and 2grams)

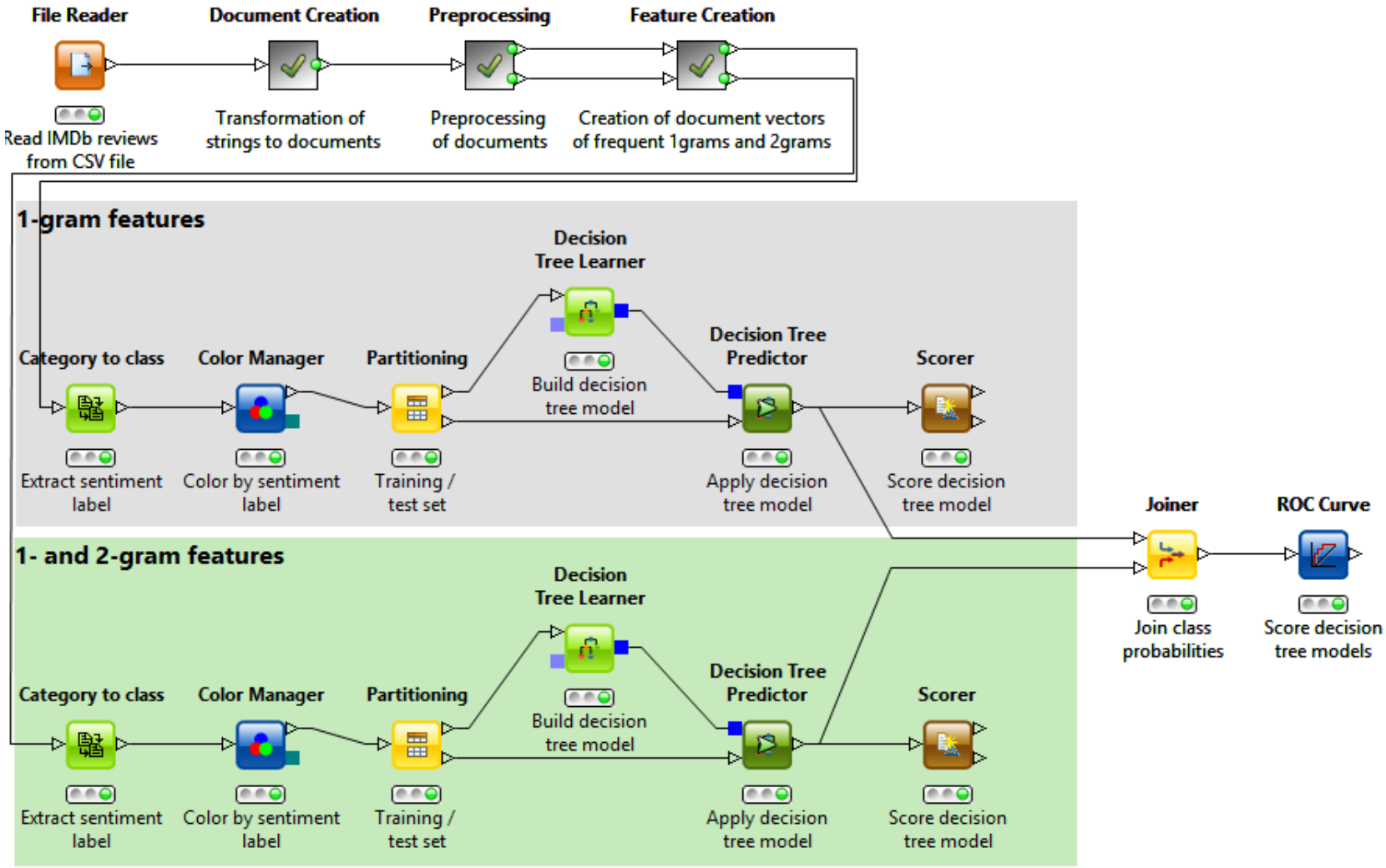
NGram Creator

- Allows for creation of
  - Character Ngrams
  - Word Ngrams
- As
  - Bag of words
  - With frequencies



S Ngram	I Cor...
special effects	318
that is	312
and it	309
that it	284
to watch	279
this one	279
the end	275
but the	274
the same	272
to make	272

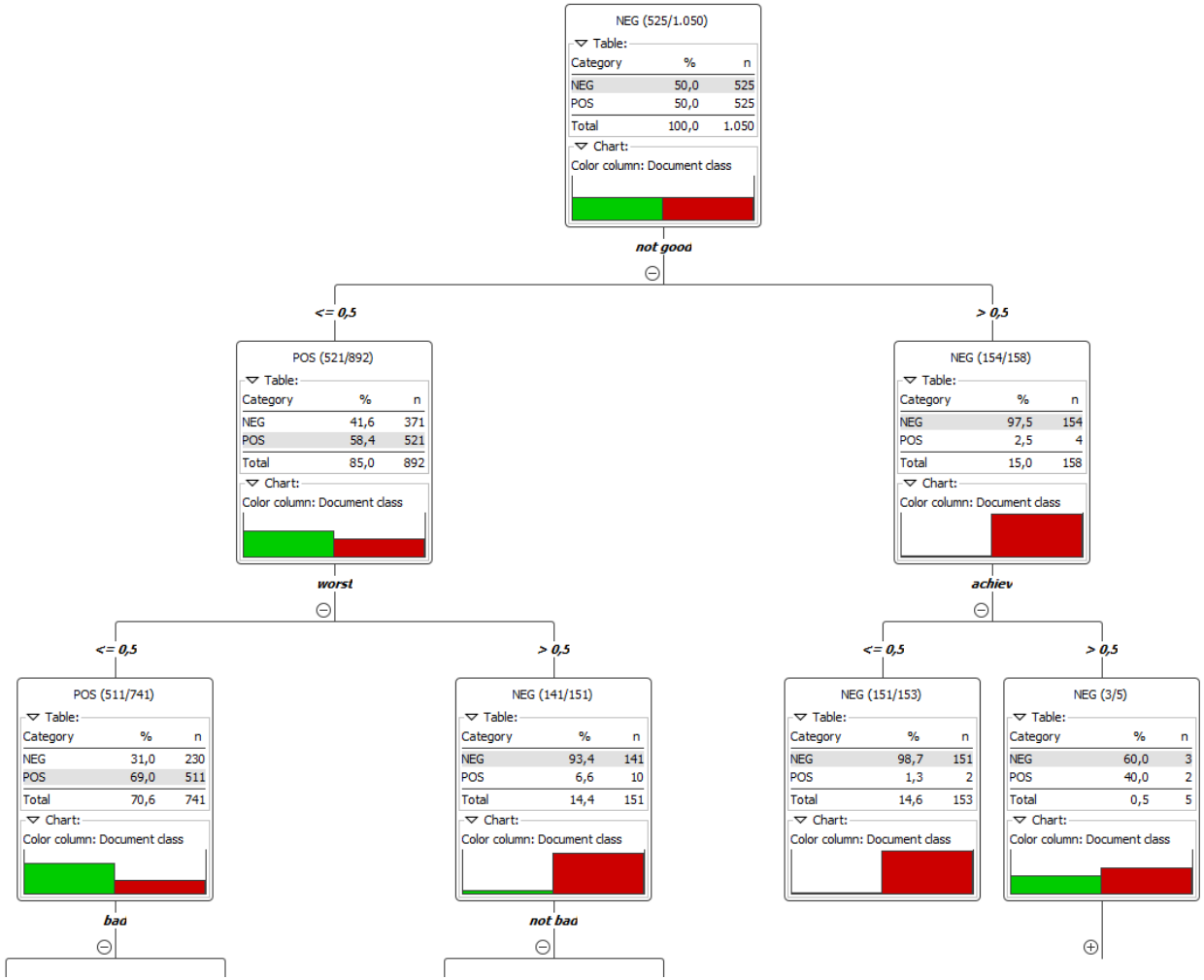
NGram Features



Hands On

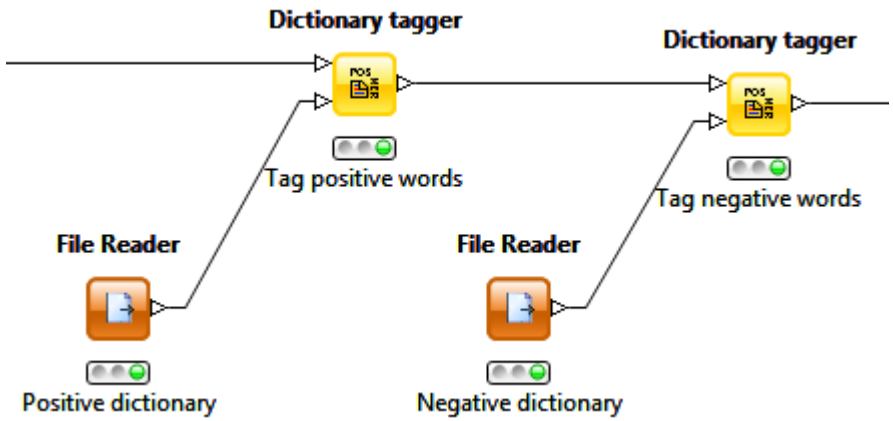
- Read movie review data
- Convert to documents
- Preprocess documents
- Create 1- and 2-gram features
- Filter features based on frequency
- Transform documents into numerical representation
- Train and score predictive model
- View decision tree view

Decision Tree Model



Dictionaries

- Custom dictionaries for sentiment classification
  - Dictionary Tagger
  - Wildcard Tagger (regular expressions)

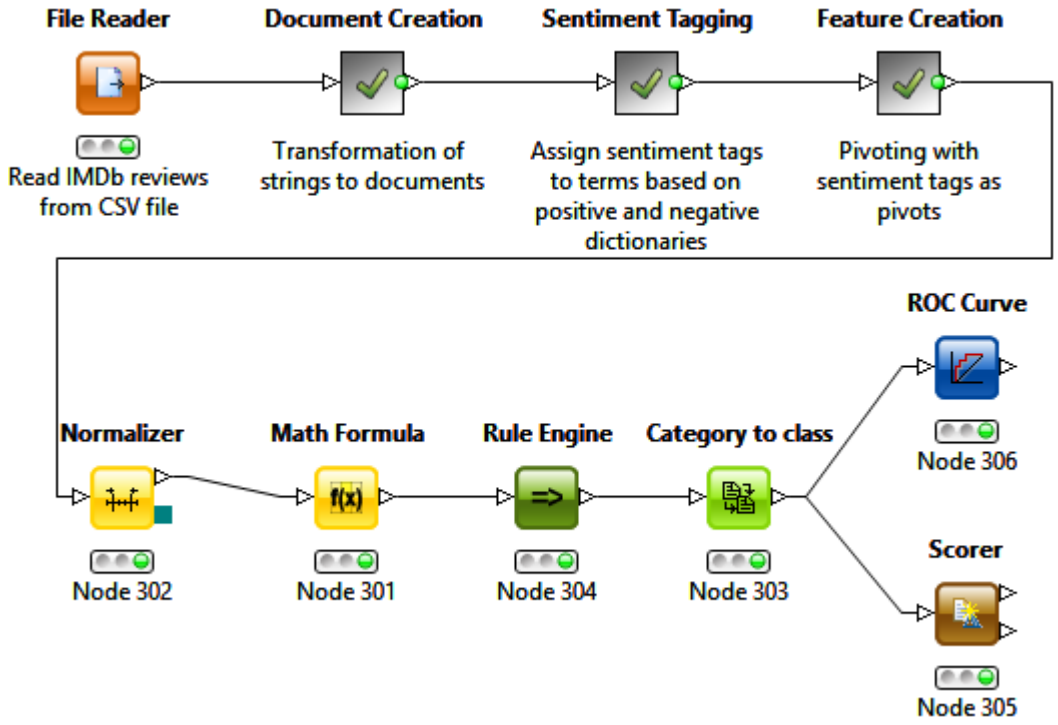


MPQA Subjectivity Lexicon

- Dictionary with words and related subjectivity clues
  - 2718 words with positive polarity
  - 4901 words with negative polarity

- http://mpqa.cs.pitt.edu/lexicons/subj_lexicon

Dictionary Based



Hands On

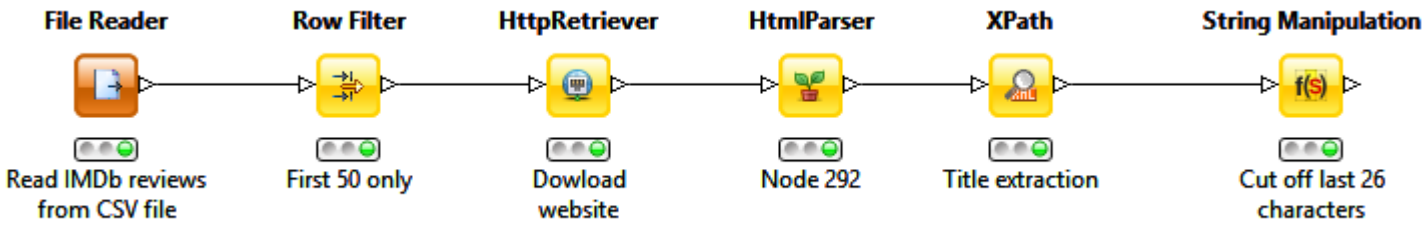
- Read movie review data
- Convert to documents
- Tag documents
- Create pivot features
- Specify score and rule for label assignment
- Score prediction

Appendix

- Palladian and XML nodes to extract move titles from IMDB web pages
- First dataset contains IMDB URLs of related movies

Index	URL
3617	http://www.imdb.com/title/tt0210075/usercomme...
3671	http://www.imdb.com/title/tt0337640/usercomme...
3157	http://www.imdb.com/title/tt0303549/usercomme...
660	http://www.imdb.com/title/tt0716825/usercomme...
265	http://www.imdb.com/title/tt0182225/usercomme...
4027	http://www.imdb.com/title/tt0347779/usercomme...
5820	http://www.imdb.com/title/tt0095655/usercomme...
1574	http://www.imdb.com/title/tt0298131/usercomme...
10668	http://www.imdb.com/title/tt0088915/usercomme...
1473	http://www.imdb.com/title/tt0828154/usercomme...
8337	http://www.imdb.com/title/tt0110099/usercomme...
11217	http://www.imdb.com/title/tt0098492/usercomme...

Web Data Extraction



Hands On

- Read movie review data
- Load webpage content
- Parse HTML
- Extract title field

Open for Innovation
KNIME

The End

Kilian Thiel
Kilian.Thiel@knime.com
@KNIME