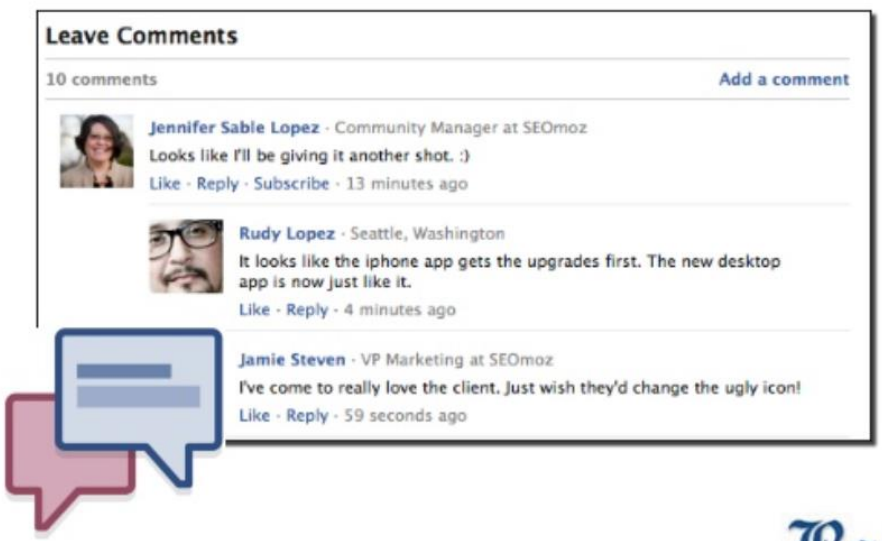
TEXT MINING

2143488 BIG DATA AND ARTIFICIAL INTELLIGENCE

DR. JING TANG

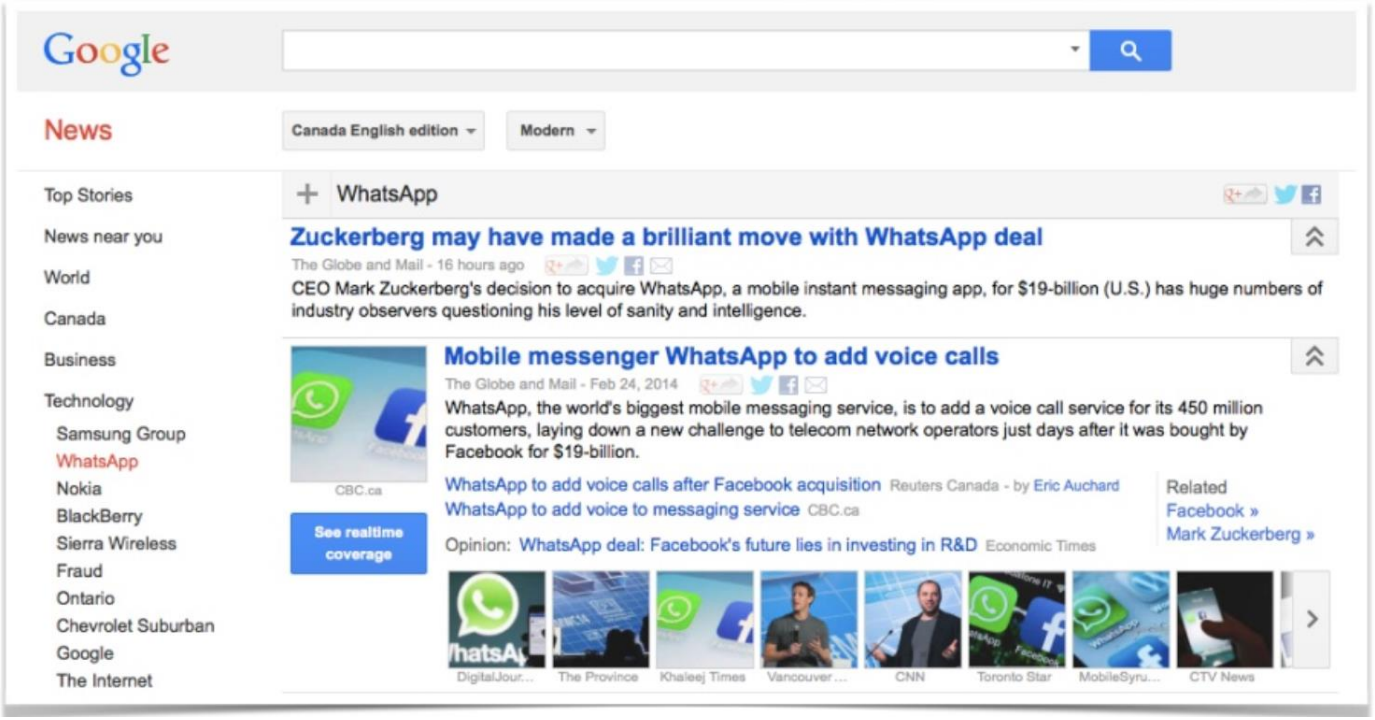
WHY TEXT IS IMPORTANT?

- Social networks (e.g., Facebook, Twitter, Instagram)
- Documents (e.g., email, SMS, reports)
- News (e.g., e-newspapers, CNN, Google News)

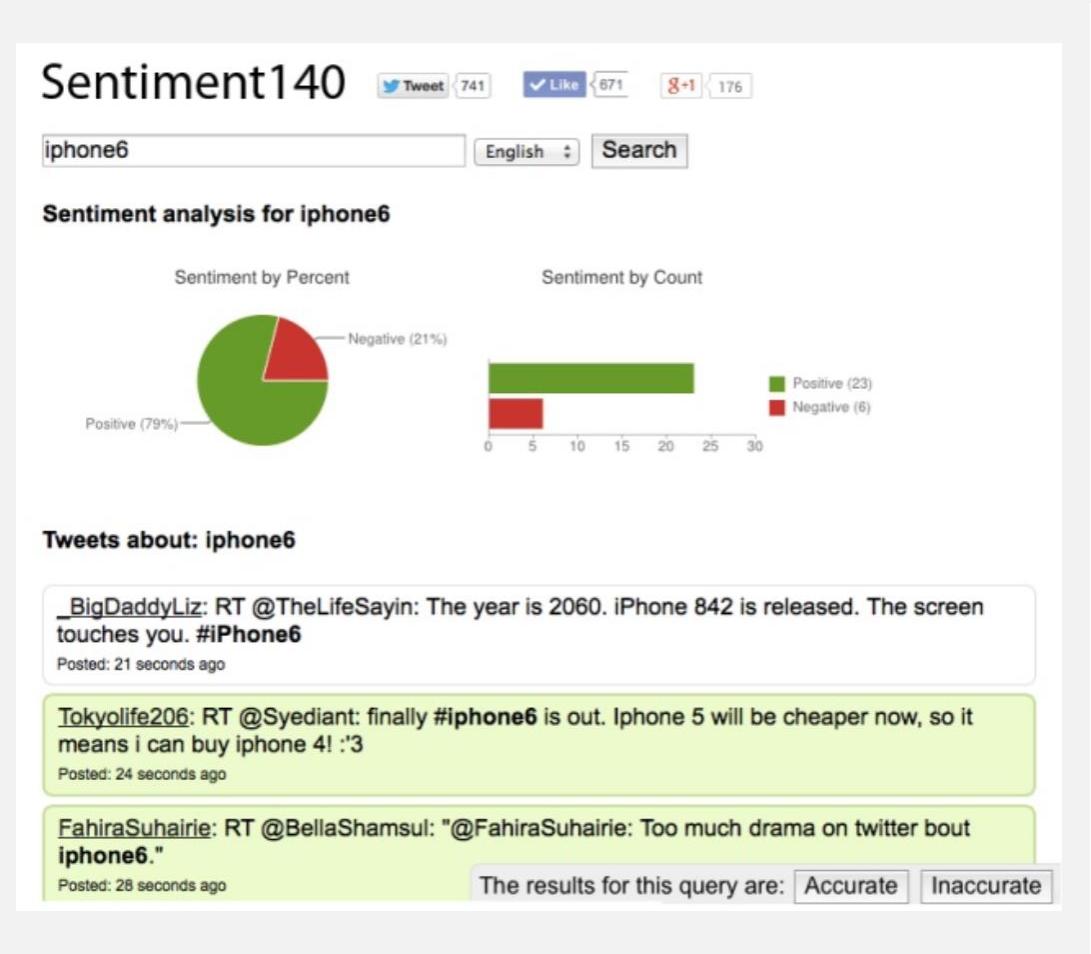


APPLICATIONS

- Find webpage or news that have <u>similar content</u> to given <u>keywords</u>



APPLICATIONS: SENTIMENT ANALYSIS



TEXT REPRESENTATION: CONVERT UNSTRUCTURED DATA TO STRUCTURED DATA

CONVERT UNSTRUCTURED DATA TO STRUCTURED DATA (1)

- Frequency of words in texts

ID	Tweet	Sentiment
1	Laying on the sand sun hitting me, this feels good.	positive
2	Smells so good I have two.	positive
3	i watched 3 movies today, reminded me of my child hood! Finding Nemo was always my favourite, but that was when i was MUCH younger!	positive
4	I think NYC would be amazing, I really wanna go there.	positive
5	he never care,he never look at me.	negative
6	i am so tired!! and my feet hurt.	negative

Term	Count
Finding	1

CONVERT UNSTRUCTURED DATA TO STRUCTURED DATA (2)

- Frequency of words in texts

ID	Tweet	Sentiment
1	Laying on the sand sun hitting me, this feels good.	positive
2	Smells so good I have two.	positive
3	i watched 3 movies today, reminded me of my child hood! Finding Nemo was always my favourite, but that was when i was MUCH younger!	positive
4	I think NYC would be amazing, I really wanna go there.	positive
5	he never care,he never look at me.	negative
6	i am so tired!! and my feet hurt.	negative

Term	Count
Finding	1
I	3

CONVERT UNSTRUCTURED DATA TO STRUCTURED DATA (3)

- Frequency of words in texts

ID	Tweet	Sentiment
1	Laying on the sand sun hitting me, this feels good.	positive
2	Smells so good I have two.	positive
3	i watched 3 movies today, reminded me of my child hood! Finding Nemo was always my favourite, but that was when i was MUCH younger!	positive
4	I think NYC would be amazing, I really wanna go there.	positive
5	he never care,he never look at me.	negative
6	i am so tired!! and my feet hurt.	negative

Bag of words

Term	Count	Term	Count
Finding	1	child	1
I	3	favourite	1
Laying	1	feels	1
MUCH	1	feet	1
NYC	1	go	1
Nemo	1	good	2
Smells	1	have	1
always	1	he	2
am	1	hitting	1
amazing	1	hood	1
and	1	hurt	1
at	1	i	3
be	1	look	1
but	1	me	1
care	1	…	1

CONVERT UNSTRUCTURED DATA TO STRUCTURED DATA (4)

- Convert the words into roots (e.g., finding → find)

Term	Count
Finding	1
I	3
Laying	1
MUCH	1
NYC	1
Nemo	1
Smells	1
always	1
am	1
amazing	1
and	1
at	1
be	1
but	1
care	1

Bag of words

Term	Count
find	1
i	6
lai	1
much	1
nyc	1
nemo	1
smell	1
alwai	1
am	1
amaz	1
and	1
at	1
be	1
but	1
care	1

Term	Count
child	1
favourite	3
feels	1
feet	1
go	1
good	2
have	1
he	2
hitting	1
hood	1
hurt	1
i	3
look	1
me	1
…	1

Bag of words

Term	Count
child	1
favourit	3
feel	1
feet	1
go	1
good	2
have	1
he	2
hit	1
hood	1
hurt	1
~~i~~	~~3~~
look	1
me	1
…	1

CONVERT UNSTRUCTURED DATA TO STRUCTURED DATA (5)

- Remove stop words (common words)

Term	Count	Term	Count
find	1	child	1
i	6	favourit	3
lai	1	feel	1
much	1	feet	1
nyc	1	go	1
nemo	1	good	2
smell	1	have	1
alwai	1	he	2
am	1	hit	1
amaz	1	hood	1
and	1	hurt	1
at	1		
be	1	look	1
but	1	me	1
care	1	…	1

Term	Count	Term	Count
find	1	hood	1
i	6	hurt	1
lai	1	look	1
nyc	1	care	1
nemo	1	movi	1
smell	1	reali	1
alwai	1	remind	1
amaz	1	sand	1
child	1	sun	1
favorit	1	thi	1
feel	1	think	1
feet	1	tire	1
go	1	todai	1
good	2	wa	3
hit	1	watch	1

CONVERT UNSTRUCTURED DATA TO STRUCTURED DATA (6)

- Binary occurrence

ID	Tweet	Sentiment
1	Laying on the sand sun hitting me, this feels good.	positive
2	Smells so good I have two.	positive
3	i watched 3 movies today, reminded me of my child hood! Finding Nemo was always my favourite, but that was when i was MUCH younger!	positive
4	I think NYC would be amazing, I really wanna go there.	positive
5	he never care,he never look at me.	negative
6	i am so tired!! and my feet hurt.	negative

attribute | label

ID	find	I	lai	nyc	nemo	smell	alwai	…	Sentiment
1	0	0	1	0	0	0	0	…	positive
2	0	1	0	0	0	1	0	…	positive
3	1	1	0	0	1	0	1	…	positive
4	0	1	0	1	0	0	0	…	positive
5	0	0	0	0	0	0	0	…	negative
6	0	1	0	0	0	0	0	…	negative

WORD IMPORTANT MEASURES (1)

- **Term frequency (TF)**
  - TF(t) = word t frequency / all words in a message

$1/6 = 0.17$

ID	Tweet	Sentiment
1	Laying on the sand sun hitting me, this feels good.	positive
2	Smells so good I have two.	positive
3	i watched 3 movies today, reminded me of my child hood! Finding Nemo was always my favourite, but that was when i was MUCH younger!	positive
4	I think NYC would be amazing, I really wanna go there.	positive
5	he never care,he never look at me.	negative
6	i am so tired!! and my feet hurt.	negative

	attribute						label
ID	find	I	lai	nyc	nemo	…	Sentiment
1	0	0	0.17	0	0	…	positive
2	0	0.17	0	0	0	…	positive
3	0.17	0.33	0	0	0.17	…	positive
4	0	0.33	0	0.17	0	…	positive
5	0	0	0	0	0	…	negative
6	0	0.17	0	0	0	…	negative

WORD IMPORTANT MEASURES (2)

- **Inverse document frequency (IDF)**
  - Measure how <u>important</u> a term is $\mathrm{IDF(t)} = \log_{10} \frac{\mathrm{N}}{\mathrm{n}}$

← attribute →									← label →
ID	find	I	lai	nyc	nemo	smell	alwai	…	Sentiment
1	0	0	1	0	0	0	0	…	positive
2	0	1	0	0	0	1	0	…	positive
3	1	1	0	0	1	0	1	…	positive
4	0	1	0	1	0	0	0	…	positive
5	0	0	0	0	0	0	0	…	negative
6	0	1	0	0	0	0	0	…	negative

N =Total number of documents

n =Total number of documents with term t

$$\mathrm{IDF(find)} = \log_{10}\left(\frac{6}{1}\right)$$

$$= \log_{10} 6 = 0.78$$

term	find	I	lai	nyc	nemo	smell	alwai
IDF	0.78	0.18	0.78	0.78	0.78	0.78	0.78

WORD IMPORTANT MEASURES (3)

- **Term frequency - Inverse document frequency (TF-IDF)**
  - A measure of how much <u>information</u> the word provides

TF table

ID	find	I	lai	nyc	nemo	…	Sentiment
1	0	0	0.17	0	0	…	positive
2	0	0.17	0	0	0	…	positive
3	0.17	0.33	0	0	0.17	…	positive
4	0	0.33	0	0.17	0	…	positive
5	0	0	0	0	0	…	negative
6	0	0.17	0	0	0	…	negative

term	find	I	lai	nyc	nemo
TF-IDF	0.000	0.000	0.133	0.000	0.000
	0.000	0.031	0.000	0.000	0.000
	0.133	0.059	0.000	0.000	0.133
	0.000	0.059	0.000	0.133	0.000
	0.000	0.000	0.000	0.000	0.000
	0.000	0.031	0.000	0.000	0.000

term	find	I	lai	nyc	nemo	smell	alwai
IDF	0.78	0.18	0.78	0.78	0.78	0.78	0.78

$$\text{TF-IDF} = \text{TF} \times \text{IDF}$$

$$\text{TF-IDF(find in doc. 3)} = 0.17 \times 0.78 = 0.133$$

EXAMPLE 1:

- When a 100-word document A contains the term “cat” 12 times. When a 100-word document B contains the term “cat” 6 times. When a 100-word document C contains the term “cat” 0 times. When a 100-word document D contains the term “cat” 0 times. Please calculate the TF-IDF of the term ”cat” in each document.

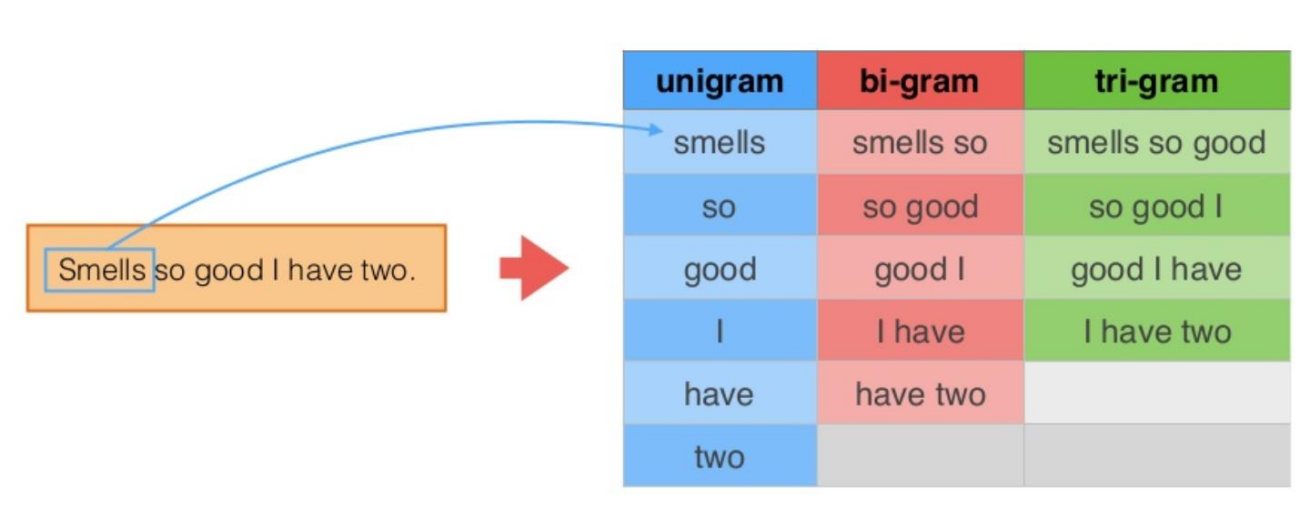
TF-IDF

- **High** TF-IDF refers to a **high rarity** of the term.
- If TF-IDF applied to a search engine, it brings benefits as:
  - stop worrying about using the **stop-words**
    - Stop-words: the most common used but unimportant words
  - successfully hunt words with **higher search volumes** and **lower competition**
  - be sure to have words that make your content **unique and relevant** to the user, etc

Sample text with Stop Words	Without Stop Words
GeeksforGeeks – A Computer Science Portal for Geeks	GeeksforGeeks , Computer Science, Portal ,Geeks
Can listening be exhausting?	Listening, Exhausting
I like reading, so I read	Like, Reading, read

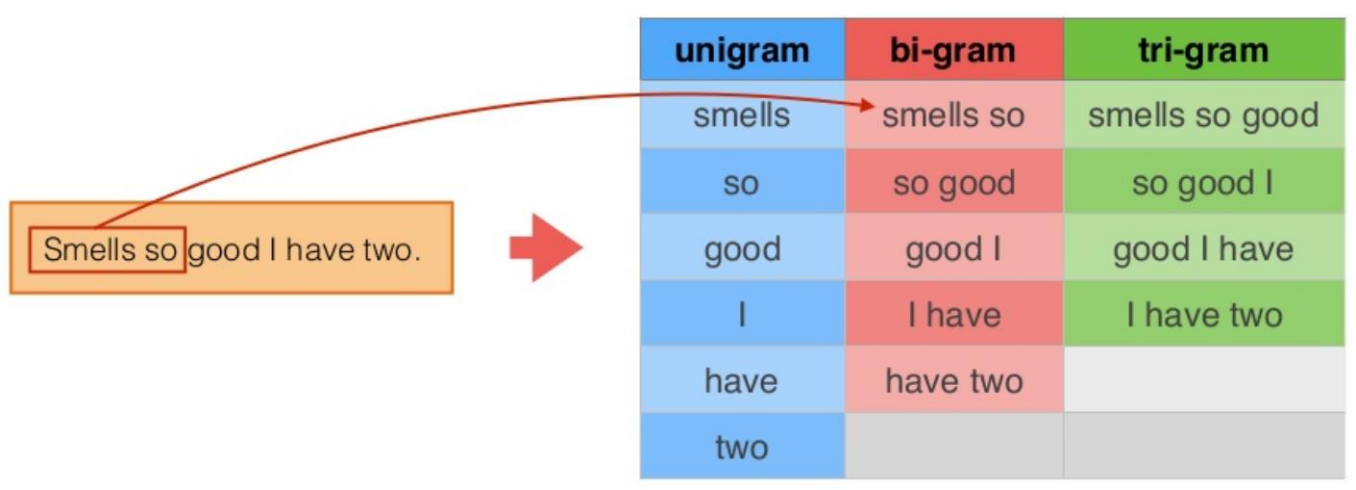
WORD IMPORTANT MEASURES (4)

- Analysis of sequential words
  - Unigram (1 words)
  - Bi-gram (2 words)
  - Tri-gram (3 words)



WORD IMPORTANT MEASURES (5)

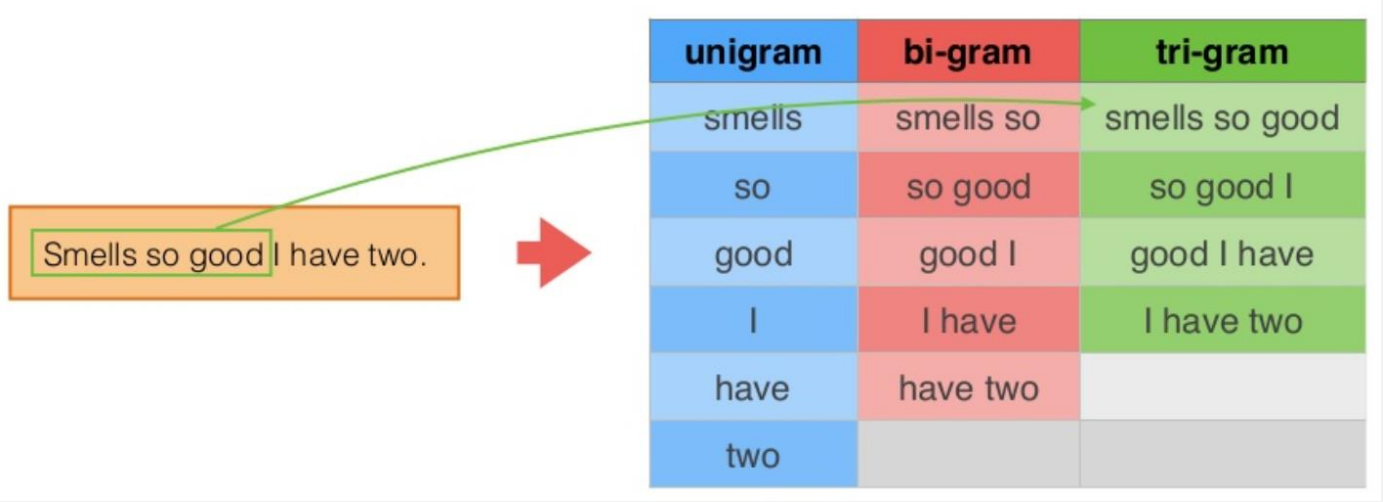
- Analysis of sequential words
  - Unigram (1 words)
  - Bi-gram (2 words)
  - Tri-gram (3 words)



unigram	bi-gram	tri-gram
smells	smells so	smells so good
so	so good	so good I
good	good I	good I have
I	I have	I have two
have	have two	
two		

WORD IMPORTANT MEASURES (6)

- Analysis of sequential words
  - Unigram (1 words)
  - Bi-gram (2 words)
  - Tri-gram (3 words)

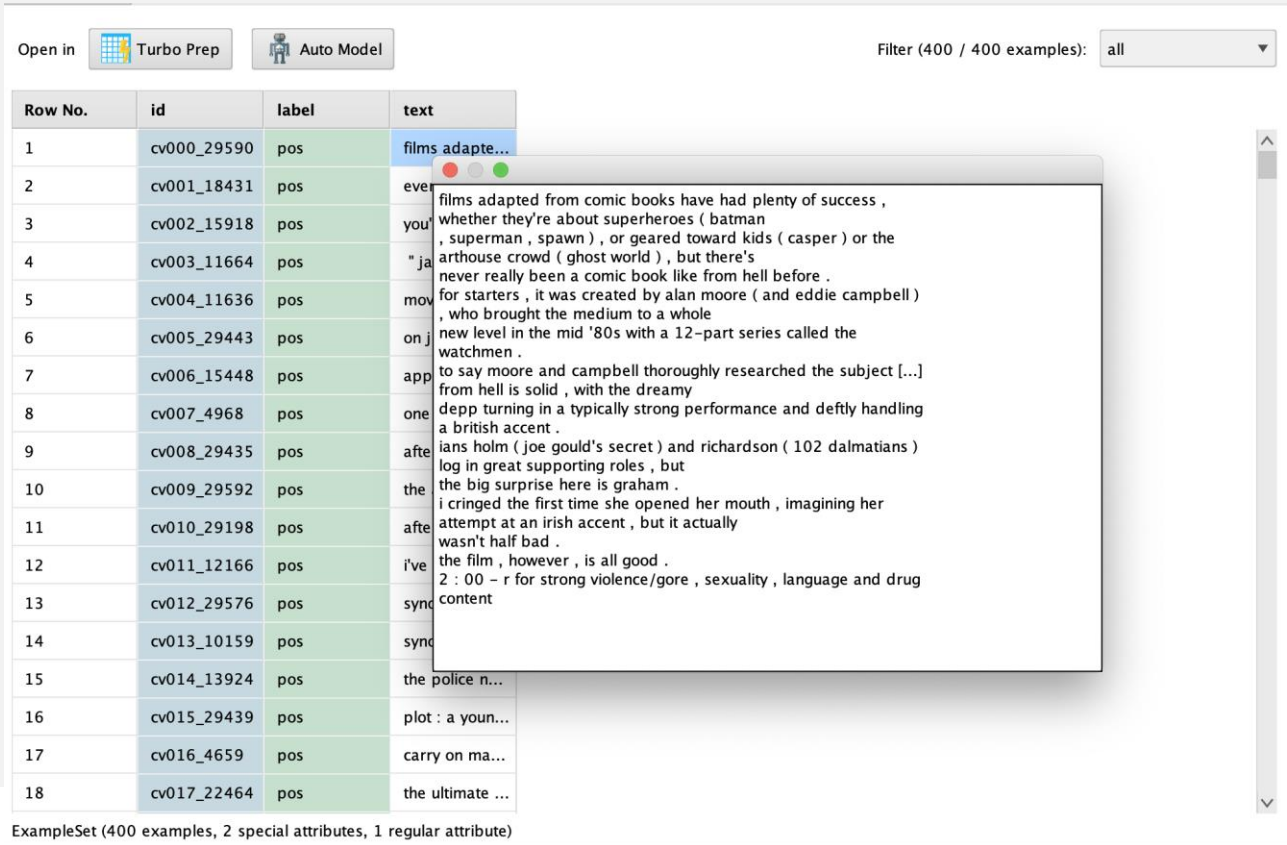


N-GRAMS

- Identify words and phrases
- Reveal relationships between words

EXAMPLE: SENTIMENT ANALYSIS OF MOVIE REVIEWS

- 400 Reviews (community samples → community dataset → entertainment → sentiments of 4000 move reviews)



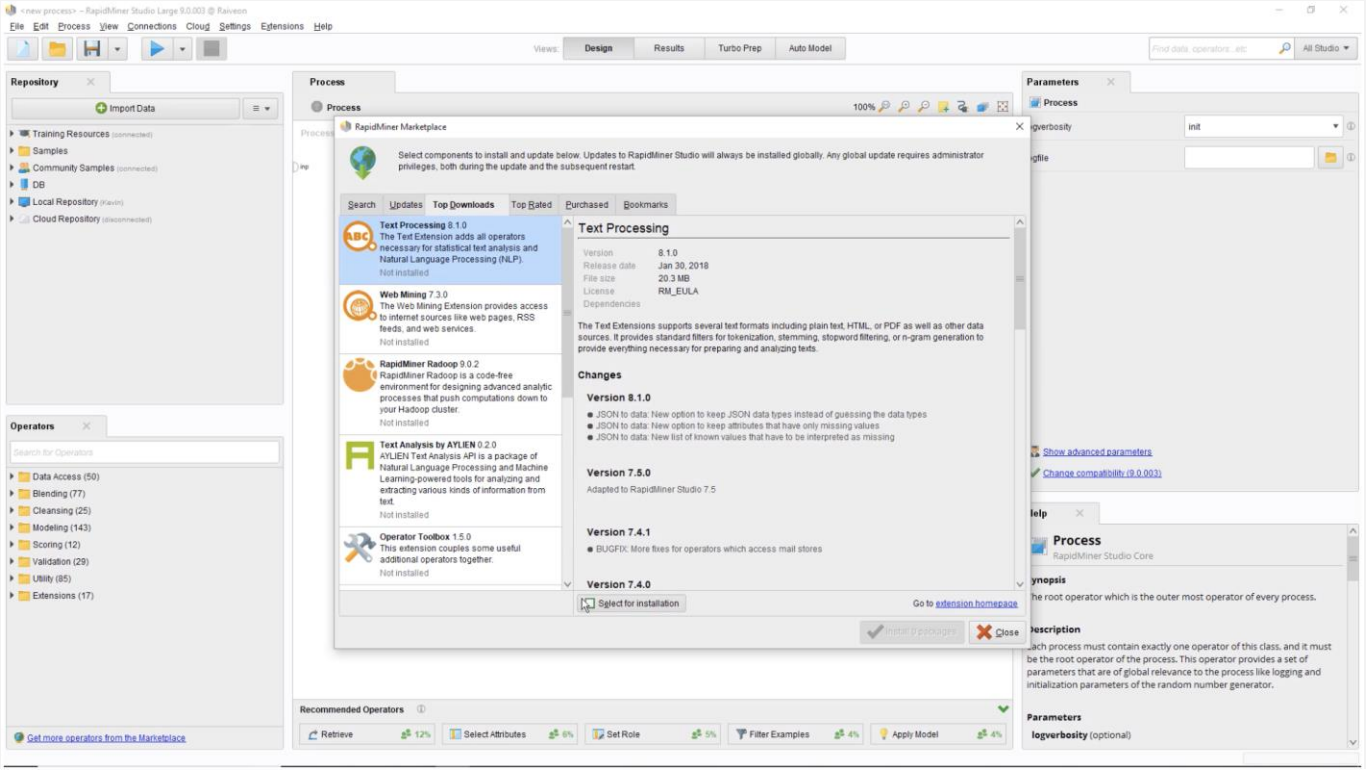
ANALYSIS STEPS

1. Data Preparation & Cleaning
   - Import data
   - Split data
   - TF_IDF Calculation (Text Vectorization)
2. Data Visualization & Analysis
   - None
3. Deep Learning Training:
   - SVM (any type of classification)
4. Testing & Evaluation:

   Confusion Matrix, Accuracy

TEXT PROCESSING EXTENSION IN RAPIDMINER

- https://academy.rapidminer.com/learn/video/loading-text-into-rapidminer



DATA PREPROCESSING

- Transform cases
- Tokenize
- Filter Tokens
- Filer Stopwords
- Stemming (Porter)

HW_14: SENTIMENT PREDICTION BASED ON US AIRLINE TWEETS

- A sentiment analysis of 14,427 tweets about US airline

1	Tweet_id	5	Negativereason_confidence
2	Airline_sentiment	6	Airline
3	Airline_sentiment_confidence	7	Test
4	Negativereason		

ANALYSIS STEPS

1. Data Preparation & Cleaning
   - Import data
   - Select attributes (id, sentiment, and text)
   - Split data
   - Text Preprocessing (text vectorization, or following page 24)
2. Data Visualization & Analysis
   - Airline Ranking based on Num. of positive and negative tweets, etc.
3. Deep Learning Training:
   - SVM (any type of classification)
4. Testing & Evaluation:

   Confusion Matrix, Accuracy